

Technische Dokumentation zur Basis- spezifikation für das QS-Verfahren am- bulante Psychotherapie

Spezifikationsjahr 2027

Informationen zum Bericht

BERICHTSDATEN

Technische Dokumentation zur Basisspezifikation für das QS-Verfahren ambulante Psychotherapie. Spezifikationsjahr 2027

Datum der Abgabe 15. September 2026

DATENQUELLEN UND BETRACHTETE ZEITRÄUME

QS-Dokumentationsdaten Spezifikationsjahr 2027

Inhaltsverzeichnis

Tabellenverzeichnis.....	8
Abbildungsverzeichnis.....	11
Abkürzungsverzeichnis	13
Änderungsindex	17
Leseanleitung	18
1 Einleitung	19
1.1 Spezifikationsbegriff	20
1.2 Zielsetzung und Zielgruppe	21
1.3 Releaseplanung.....	22
A Prozesse	27
1 QS-Dokumentation.....	29
1.1 Auslösung	30
1.1.1 Der QS-Filter-Eingangsdatensatz.....	31
1.1.2 Besonderheiten im QS-Verfahren ambulante Psychotherapie	31
1.2 Erfassung.....	32
1.2.1 Gestaltung von Eingabemasken.....	32
1.2.2 Einrichtungsidifizierende Daten	36
1.2.3 QS-Daten	40
1.2.4 Plausibilitätsprüfungen.....	41
1.3 Export der Daten aus der QS-Dokumentation	43
1.3.1 Erzeugen der Exportdatei	44
1.3.2 Datenprüfung	47
1.3.3 XML-Verschlüsselung	47
1.3.4 Ausgangvalidierung	48
1.3.5 Beispiele für Exportdateien	49
1.4 Datenübermittlung	49
1.4.1 Dateibenennung	49
1.4.2 Datenversand via E-Mail – Krankenhäuser	50

1.4.3	Datenversand via gesicherter Schnittstellen – Arztpraxen/MVZ für ambulant-kollektivvertraglich erbrachte Leistungen	50
1.4.4	Datenübermittlung für selektivvertraglich erbrachte Leistungen (Arztpraxen/MVZ)	50
1.4.5	Zusammenfassung Datenversand	51
1.5	Rückprotokollierung	51
1.6	Datenannahme	52
1.6.1	Datenübertragung und Authentifizierung	52
1.6.2	Fristen für die Datenannahme und -weiterleitung	53
1.7	Datenprüfung	53
1.7.1	Dateiprüfungen	54
1.8	Pseudonymisierung der LE-Daten	55
1.9	Weitere Datenverarbeitungen	57
1.10	XML-Verschlüsselung	58
1.11	Datenübermittlung	58
1.11.1	Ausgangsprüfung	59
1.11.2	Datenweiterleitung	59
1.12	Rückprotokollierung durch die Datenannahmestellen	60
1.12.1	Empfangsbestätigung	60
1.12.2	Miniprotokoll	60
1.12.3	Datenflussprotokoll	60
1.12.4	Empfangsbestätigung durch die VST	60
1.13	Weiterleitung der Datenflussprotokolle der BAS	61
1.14	Zusammenfassung	62
2	Risikostatistik	66
3	Sollstatistik	67
3.1	Konformitätserklärung	68
4	Auswertungen	70
5	Allgemeine Regelungen zur Datenübermittlung	73
5.1	Datenfluss	73
5.1.1	Datenfluss der QS-Daten	73

5.1.2	Datenfluss der Rückprotokolle.....	75
5.2	Datenübermittlung	76
5.2.1	Gesicherte Datenübertragung	76
5.2.2	Abgrenzung von Test-, Probe- und Regelbetrieb	86
5.3	Rückprotokollierung	90
5.3.1	Funktion von Empfangsbestätigung und Datenflussprotokoll im Datenfluss	90
5.3.2	Die Rückprotokollierung.....	92
5.4	Meldung eines Leistungserbringermappings.....	100
B	Komponenten.....	101
1	QS-Filter	106
1.1	Anmerkungen zur Struktur der Spezifikationsdatenbank für QS-Filter.....	106
1.2	Grundlegende Tabellen der Datenbank.....	107
1.2.1	Module (Datensätze der QS-Dokumentation)	107
1.2.2	Struktur der Datensatzdefinitionen.....	108
1.2.3	Funktionen.....	112
1.2.4	ICD-Listen.....	113
1.2.5	FAB-Listen.....	113
1.2.6	EBM-Listen	113
1.2.7	Versionsverwaltung	114
1.2.8	Meta-Tabellen	116
1.2.9	DB-Änderungen gegenüber der Vorgängerversion	116
1.3	Der QS-Filter-Datensatz.....	116
1.3.1	Der QS-Filter-Eingangsdatensatz.....	117
1.3.2	Der Datensatz „Anwenderbezogene Fallidentifikation“	123
1.4	Der Algorithmus zur Ermittlung der Dokumentationspflicht	123
1.4.1	Einleitung und Überblick.....	123
1.4.2	Leistungsbereichsbezogene Einschlusskriterien.....	124
1.4.3	Administrative Einschlusskriterien	127
1.4.4	Struktur und Syntax der Auslösebedingungen	128
1.4.5	Stufen der Dokumentationsverpflichtung	131

1.4.6	Fehlerprüfung	132
1.5	Sollstatistik	134
2	QS-Dokumentation.....	135
2.1	Patientenidentifizierende Daten zur Follow-up-Erhebung	135
2.2	Datenfelddbeschreibung.....	135
2.2.1	Dokumentationsmodule (Datensätze)	137
2.2.2	Teildatensätze	138
2.2.3	Datenfelder (Bogenfelder)	138
2.2.4	Überschriften	147
2.2.5	Ausfüllhinweise	147
2.3	Plausibilitätsprüfungen.....	149
2.3.1	Die Regeltabelle	149
2.3.2	Regelsyntax	151
2.3.3	Funktionen.....	157
2.3.4	Syntaxvariablen.....	159
2.3.5	Einzelregeln	160
2.3.6	Feldgruppenregeln	160
2.3.7	Prüfung von Feldeigenschaften.....	168
2.3.8	Verfahren für die Evaluation von Regeln	172
2.4	Listen von Schlüsselcodes (ICD-10-GM).....	173
2.4.1	OPS-Listen.....	174
2.4.2	ICD-Listen.....	174
2.4.3	Relationstabellen ICD- Listen	174
2.5	Exportfelddbeschreibung	174
2.5.1	Exportmodule.....	175
2.5.2	Exportdatensatz	176
2.6	Versionierung.....	181
2.6.1	Grundlegende Definitionen.....	181
2.6.2	Delta-Informationen zur vorhergehenden Version.....	182
2.6.3	Abgrenzung zwischen Spezifikationsjahren und Datensatzformaten.....	185
2.6.4	Version des Exportverfahrens	186

2.7	Administrative Objekte	186
2.7.1	CSV/XML-Mapping in der Spezifikationsdatenbank (QSDOK)	186
2.7.2	Datenservices	190
2.7.3	Prüfschritte	191
3	Risikostatistik	194
4	XML-Schema	195
4.1	Kompositionsmodell	195
4.2	Schnittstellen	196
4.3	Darstellung der XML-Struktur	197
4.4	Aufbau der XML-Exportdatei	198
4.4.1	Namensräume	198
4.4.2	Wurzelement <root>	199
4.4.3	Header-Bereich	200
4.4.4	Body-Bereich	212
4.5	Umgang mit Umlaut-Domains und E-Mail-Adressen	226
5	Tools	228
5.1	Datenprüfprogramm	228
5.1.1	Umfang der Prüfungen	228
5.1.2	Ausgangskontrolle vor Versand	228
5.1.3	Programmaufruf	229
5.1.4	Verzeichnisstruktur	233
5.1.5	Ausgabe	233
5.1.6	Grafische Oberfläche	234
5.1.7	Programmierschnittstelle – API	235
5.2	Verschlüsselungspaket	235
5.3	LE-Pseudonymisierungsprogramm	235
C	Anhang	236
	Glossar	236
	Impressum	241

Tabellenverzeichnis

Tabelle 1: Reguläre Meilensteine der Releaseplanung der Spezifikation für den Regelbetrieb	23
Tabelle 2: Übersicht über die Prozesse QS-Dokumentation, Sollstatistik und Risikostatistik und ihre Unterprozesse.....	27
Tabelle 3 Informationen aus der Datenbank, welche im GUI verwendet werden	32
Tabelle 4: Adressdaten im Modul PAPSYP	39
Tabelle 5: Arten der Plausibilitätsprüfungen.....	42
Tabelle 6: Asymmetrische Verschlüsselung der XML-Elemente.....	48
Tabelle 7: XML-Schemata zur Prüfung nach der Verschlüsselung	48
Tabelle 8: XML-Beispiele.....	49
Tabelle 9: Zuständige Datenannahmestelle.....	51
Tabelle 10: Pseudonymisierung der LE-Daten innerhalb des Elements <code>care_provider</code> (<code>DeQS-RL</code>).....	56
Tabelle 11: Kodierung der Landesgeschäftsstellen der KV	57
Tabelle 12: Bisher vorgesehene Datenübertragungswege	59
Tabelle 13: Aufgaben der Leistungserbringer in Bezug auf die QS-Dokumentation.....	63
Tabelle 14: Aufgaben der Datenannahmestellen.....	64
Tabelle 15: Dateinamensbestandteile von Auslieferungseinheiten und deren beinhaltete Dateien	70
Tabelle 16: Benennungselemente der Exportdateien.....	79
^^Tabelle 17: Ausfüllen der Elemente eines Validation-Items in Abhängigkeit von den Fehlerarten	94
Tabelle 18: Beispiel für eine Fehlermeldungen	95
Tabelle 19: Struktur der Tabelle Modul	108
Tabelle 20: Struktur der Tabelle Ds	109
Tabelle 21: Struktur der Tabelle Tds.....	109
Tabelle 22: Struktur der Tabelle BasisTyp.....	110
Tabelle 23: Struktur der Tabelle Schluessel	111
Tabelle 24: Struktur der Tabelle SchluesselWert	112
Tabelle 25: Struktur der Tabelle Version	114
Tabelle 26: Ausschnitt der Tabelle Ds	117
Tabelle 27: Ausgewählte Felder des QS-Filter-Eingangsdatensatzes nach § 301.....	118
Tabelle 28: Ausgewählte Felder des QS-Filter-Eingangsdatensatzes nach § 295.....	120
Tabelle 29: Felder des QS-Filterdatensatzes "Anwenderbezogene Fallidentifikation"	123
Tabelle 30: Struktur der Tabelle ModulAusloeser	126
Tabelle 31: Struktur der Tabelle AdminKriterium	128
Tabelle 32: Basistypen der Variablen	129
Tabelle 33: Präzedenz und Assoziativität der Operatoren.....	130

Tabelle 34: Stufen der Dokumentationsverpflichtung	132
Tabelle 35: Fehlercodes des QS-Filters	132
Tabelle 36: Struktur der Tabelle Modul	137
Tabelle 37: Struktur der Tabelle BogenFeld	138
Tabelle 38: Struktur der Tabelle Feld	141
Tabelle 39: Struktur der Tabelle BasisTyp	143
Tabelle 40: Struktur der Tabelle Schluessel	144
Tabelle 41: Struktur der Tabelle SchluesselWert	146
Tabelle 42: Schlüssel mit spezieller Sortierung (pTMamma)	147
Tabelle 43: Struktur der Tabelle Abschnitt	147
Tabelle 44: Arten von Hinweistypen (Spezifikation 2018)	148
Tabelle 45: Tabelle RegelTyp	149
Tabelle 46: Struktur der Tabelle Regeln	149
Tabelle 47: Struktur der Tabelle RegelFelder	150
Tabelle 48: Struktur der Tabelle MehrfachRegel	151
Tabelle 49: Basistypen der Datenfelder in den Plausibilitätsregeln	152
Tabelle 50: Präzedenz und Assoziativität der Operatoren	155
Tabelle 51: Typen von Feldgruppen	161
Tabelle 52: Struktur der Tabelle FeldGruppe	162
Tabelle 53: Struktur der Tabelle FeldgruppeFelder	163
Tabelle 54: Formale Definition einer Feldgruppe	164
Tabelle 55: Beispiele für die Evaluation von Regeln	173
Tabelle 56: Struktur der Tabelle ExportModul	175
Tabelle 57: Struktur der Tabelle ZusatzFeld	178
Tabelle 58: Struktur der Tabelle Ersatzfeld	179
Tabelle 59: Struktur der Tabelle ErsatzFuerFeld	179
Tabelle 60: Struktur der Tabelle DeltaNeu	183
Tabelle 61: Struktur der Tabelle DeltaAttribut	183
Tabelle 62: Struktur der Tabelle DeltaGeloescht	184
Tabelle 63: Inhalt der Tabelle TabellenFeldStruktur (fkTabellenFeldStruktur = Regeln)	185
Tabelle 64: Struktur der Tabelle ExportFormatExportModul	189
Tabelle 65: Felder der Abfrage vPruefung	191
Tabelle 66: XML-Schemata für die Rückprotokollierung	197
Tabelle 67: Symbole in den XML-Schema-Diagrammen	197
Tabelle 68: Root-Element – Attribute	200
Tabelle 69: Kind-Elemente des Elements document	201
Tabelle 70: Kind-Elemente des Elements software	203
Tabelle 71: Kind-Elemente des Elements information_system	203

Tabelle 72: Angabe des betreffenden Datenflusses	204
Tabelle 73: Angabe des betreffenden Datenfluss-Ziels	204
Tabelle 74: Attribute des Elements header/provider	205
Tabelle 75: Attribute des Elements header/protocol	206
Tabelle 76: Attribute des Elements validation_provider	207
Tabelle 77: Attribute des Elements validation_item	208
Tabelle 78: Attribute des Elements status_document	209
Tabelle 79: Attribut des Elements status	209
Tabelle 80: Attribut des Elements error	210
Tabelle 81: Kind-Elemente des Elements error	210
Tabelle 82: Leistungserbringeridentifizierende Daten im kollektiven, selektiven und stationären Bereich	213
Tabelle 83: Attribute des Elements cases	214
Tabelle 84: Kind-Elemente des Elements case_admin	216
Tabelle 85: Kind-Elemente des Elements patient_ngkv/et_pid	219
Tabelle 86: Kind-Elemente des Elements patient_tx	220
Tabelle 87: Kind-Element des Elements statistic	222
Tabelle 88: Attribut des Elements sent	222
Tabelle 89: Kind-Elemente des Elements statistic/sent	222
Tabelle 90: Attribute des Elements statistic	223
Tabelle 91: Kind-Elemente des Elements statistic/processed	224
Tabelle 92: Algorithmus zum Schreiben von statistic/processed in der Bundesauswertungsstelle	224

Abbildungsverzeichnis

Abbildung 1: Überblick über die Prozesse und Werkzeuge der Datenannahme der QS-Daten	29
Abbildung 2: Beispiel für Informationen, die in der Oberfläche angezeigt werden sollen (Spezifikation 2027)	34
Abbildung 3: Übersicht der Datenflüsse direkte/indirekte PID-/Nicht-PID-Verfahren.....	74
Abbildung 4: Übersicht der Datenflüsse der Rückprotokollierung (orange = Empfangsbestätigung).....	76
Abbildung 5: Übersicht über die einzusetzenden Suffixe.....	80
Abbildung 6: Datenflüsse im Test-, Probe- und Regelbetrieb am Bsp. der Follow-up- Verfahren	89
Abbildung 7: Attribut "originator" im Prüfungs- und Fehlerprotokoll	91
Abbildung 8: Beispiel einer Empfangsbestätigung.....	92
Abbildung 9: Beziehungen zwischen <validation_item> im header und <validation_item> im body über die id	97
Abbildung 10: Zuordnung der Version des QS-Filters zu den Behandlungsfällen: Kriterium ist das Aufnahmedatum	116
Abbildung 11: Tabellen und Relationen der Datenfeldbeschreibung (Beispiel Spezifikation 2017)	136
Abbildung 12: Feldgruppe NEO:OPart auf dem Dokumentationsbogen (Spezifikation 2017).....	167
Abbildung 13: Algorithmus zur Evaluation von Plausibilitätsregeln.....	173
Abbildung 14: Beziehungen der administrativen Objekte – Mapping-Informationen	187
Abbildung 15: Ergebnis der Abfrage vExportZieleXml (Auszug)	188
Abbildung 16: Beispiel für XPath-Ausdrücke in der Tabelle ExportZielXml in Verbindung mit weiteren Informationen.....	188
Abbildung 17: Tabelle Datenservice	191
Abbildung 18: Verfahrensbezogene Datenservices	191
Abbildung 19: Dateiordner der Schnittstellen-Schemata	196
Abbildung 20: Root-Element und Kind-Elemente header und body	199
Abbildung 21: Aufbau des Elements header	200
Abbildung 22: Aufbau des Elements document.....	201
Abbildung 23: Aufbau des Elements software.....	202
Abbildung 24: Aufbau des Elements information_system	203
Abbildung 25: Aufbau des Elements provider	205
Abbildung 26: Aufbau des Elements header/protocol	206
Abbildung 27: Aufbau und Kind-Elemente des Elements validation_provider.....	207
Abbildung 28: Aufbau und Kind-Elemente des Elements validation_item	208
Abbildung 29: Aufbau des Elements status_document	209
Abbildung 30: Aufbau und Kind-Elemente des Elements status	209

Abbildung 31: Aufbau des Elements error	210
Abbildung 32: Aufbau und Attribute des Elements encryption(Krankenhaus)	211
Abbildung 33: Aufbau des Elements body.....	212
Abbildung 34: Aufbau des Elements body/data_container	212
Abbildung 35: Aufbau des Elements care_provider – kollektiv-, selektivvertraglich	213
Abbildung 36: Aufbau des Elements cases	214
Abbildung 37: Aufbau des Elements case	215
Abbildung 38: Aufbau des Elements case_admin	215
Abbildung 39: Aufbau des Elements patient	217
Abbildung 40: Aufbau des Elements pid	218
Abbildung 41: Aufbau des Elements patient_ngkv	219
Abbildung 42: Aufbau des Elements patient_tx	220
Abbildung 43: Aufbau des Elements case_admin/protocol.....	221
Abbildung 44: Aufbau des Elements statistic.....	221
Abbildung 45: Aufbau des Elements sent	222
Abbildung 46: Aufbau des Elements processed	223
Abbildung 48: Diagramme „Bogen komplex“ und „Bogen einfach“	225
Abbildung 49: Ausprägungen des qs_data-Elements (Erfassungsmodule).....	226
Abbildung 50: Weiche Schemavariante für das DPP	232
Abbildung 51: Beispiel einer typischen Verzeichnisstruktur.....	233
Abbildung 52: Beispiel für eine Index.html Datei im Ordner <output>/html.....	234
Abbildung 53: Grafische Oberfläche des Datenprüfprogramms	234

Abkürzungsverzeichnis

Abkürzung	Bedeutung
AES	Advanced Encryption Standard (Verschlüsselungsalgorithmus)
AIS	Arztinformationssystem
AG	Arbeitsgruppe
AG DeQS	Arbeitsgruppe DeQS des G-BA
ASCII	American Standard Code for Information Interchange (Amerikanischer Standard-Code für den Informationsaustausch)
BAS	Bundesauswertungsstelle
BE	Bundesebene
BfArM	Bundesinstitut für Arzneimittel und Medizinprodukte
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
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
FES	fortgeschrittene elektronischen Signatur
FU	Follow-up

Abkürzung	Bedeutung
G-BA	Gemeinsamer Bundesausschuss
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 Qualitätssicherung 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
KIM	Kommunikation im Medizinwesen
KV	Kassenärztliche Vereinigung
KVDT	Kassenärztliche Vereinigung-Datentransfer (Datenformat)
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
MD	Medizinischer Dienst
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
PlanQI	planungsrelevante Qualitätsindikatoren
PR	Arztpraxis
Qb-R	Regelungen zum 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
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
TB	Testbetrieb
TDS	Teildatensatz

Abkürzung	Bedeutung
TPacker	Programm für die Transportverschlüsselung
URL	Uniform Resource Locator (einheitlicher Ressourcenzeiger)
V	Versionierung
VPB	Versendestelle Patientenbefragung
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.

Die spezifischen Änderungen der vorliegenden Technischen Dokumentation werden im Folgenden mit Bezug zur jeweiligen Version dargestellt. Kleinere Fehlerkorrekturen und Aktualisierungen sind im Folgenden nicht immer explizit benannt.

Da es sich beim vorliegenden Dokument um eine Erstversion handelt, sind noch keine Änderungen aufgeführt.

Änderung	Kapitel/Abschnitt	Version

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 beschrieben und spiegeln die reale, chronologische Abfolge wider. Jede Prozessbeschreibung berücksichtigt zudem die unterschiedlichen Komponenten, die für die Umsetzung benötigt und in Abschnitt B detailliert beschrieben werden. Im Abschnitt C 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 o-der 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.

1 Einleitung

Die Technische Dokumentation für Leistungserbringer beschreibt die Spezifikation zur einrichtungs- und sektorenübergreifenden Qualitätssicherung gemäß §§ 136 ff. SGB V. Die vorliegende Technische Dokumentation richtet sich dabei an die Leistungserbringer und an die mit der Umsetzung der Spezifikation beauftragten Softwarehersteller.

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 Leistungserbringer und Softwarehersteller relevanten Prozesse und Komponenten.

Regelungsbereich der Basisspezifikation sind die verschiedenen Verfahren gemäß der Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL), die vom Gemeinsamen Bundesausschuss (G-BA) beschlossen wird. 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.¹ Für eine spezifikationskonforme Umsetzung von QS-Softwareprodukten sind alle Spezifikationskomponenten zu berücksichtigen.

Aus Gründen der leichteren Lesbarkeit werden in der vorliegenden Technischen Dokumentation die Verfahren QS PCI und QS NET 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.

Nach § 1 Abs. 5 DeQS-RL wird wie folgt unterschieden:

- sektorspezifisch
- sektorenübergreifend
 - sektorenüberschreitend
 - sektorgleich

(zzgl. Follow-up-Verfahren)

¹ <https://iqtig.org/datenerfassung/spezifikationen/>

Die Verfahren, die sich auf den stationären Sektor beschränken (z.B. PNEU, HCH, LLS, CHE) werden im folgenden Dokument nicht als einrichtungs- und/oder sektorenübergreifende Verfahren bezeichnet.

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 Spezifikationsjahr. 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 Spezifikationsjahr zu 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 einrichtungs- und 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 (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 in der gesonderten Spezifikationskomponente „TechDok_Benennungsschema“ vorgegeben. 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.

1.2 Zielsetzung und Zielgruppe

Die Basisspezifikation ist ein komplexes Regelwerk, das mithilfe verschiedener Komponenten verbindliche Grundlagen für alle Prozesse im Zusammenhang mit der Erfassung und Übermittlung von QS-Daten bei den unterschiedlichen Verfahrensteilnehmern (Leistungserbringer, Datenannahmestellen, Vertrauensstelle) vorgibt und beschreibt. Die Komponenten der Spezifikation sind daher so ausgestaltet, dass sie von QS- und/oder IT-/EDV-Expertinnen und -Experten verstanden werden. Die Spezifikation richtet sich ausschließlich an diesen Teilnehmerkreis. Die Regelung und die Art der Darlegung der Spezifikationskomponenten sind auf eine möglichst automatisierte Nutzung durch diesen Personenkreis ausgerichtet.

Externe Qualitätssicherungsmaßnahmen, die einen Vergleich der Qualität von erbrachten Leistungen zum Ziel haben, stellen eine Reihe von Anforderungen an die Datenerhebung, Datenerfassung und Plausibilitätsprüfung, um valide, reliable und vergleichbare Daten gewinnen zu können. Die Erfassung und Plausibilitätsprüfung durch unterschiedliche Softwareumsetzungen beinhaltet grundsätzlich die Gefahr einer Verzerrung der Daten.

Die Vorgaben der Spezifikation, die eine einheitliche Festlegung von Datenfelddescriptions, Plausibilitätsregeln, Grundsätzen der Benutzerschnittstellengestaltung und Datenübermittlungsformaten umfassen, sollen dazu dienen, dieser Gefahr entgegenzuwirken. Dadurch werden die Erhebung valider und vergleichbarer Daten sowie ein unter datenschutzrechtlichen Gesichtspunkten sicherer Datenfluss gewährleistet.

Die vorliegende Technische Dokumentation erklärt einleitend, worauf sich die Spezifikation bezieht und welche Veröffentlichungstermine und Fristen für Rückmeldungen zur Basisspezifikation bestehen.

Anschließend werden alle für die Leistungserbringer bzw. ihre Softwarehersteller relevanten Prozesse beschrieben. Diese Prozessbeschreibung dient der näheren Erläuterung der erforderlichen Schritte beim Leistungserbringer und ist als verbindliche Handlungsanleitung zu betrachten. Damit soll erreicht werden, dass alle Leistungserbringer die Komponenten korrekt anwenden, Dokumentationspflichten erkennen und Klarheit darüber besteht, wie Datenlieferungen zu verschlüsseln und an welche Datenannahmestelle sie zu versenden sind.

Darüber hinaus werden in der Technischen Dokumentation die Spezifikationskomponenten aufgeführt und beschrieben. Wesentliche Komponenten sind die Datenbanken, die die Grundlage zur Erstellung der Softwareprodukte bilden. Auch die weiteren Komponenten enthalten Vorgaben für die Softwareprodukte und zur Umsetzung der Prozesse. In der Komponentenbeschreibung werden unter anderem Struktur, Funktionsweise und Inhalte der Datenbanken, des XML-Schemas und der weiteren Tools (z. B. Datenprüfprogramm und Verschlüsselungsprogramm) erläutert.

Die verbindlichen Vorgaben der Spezifikation sind einzuhalten. Die Art der Umsetzung kann jedoch individuell auf die Zielgruppen der Software ausgerichtet werden. Ein Beispiel hierfür wäre die verfahrensspezifische Zurverfügungstellung und Erläuterung der generischen technischen

Fehlermeldungen. Da beispielsweise die Fehlermeldungstexte der administrativen Prüfungen allgemein formuliert sind, können Verfahrensteilnehmer (Softwareanbieter, Datenannahmestellen, Vertrauensstelle) die Meldungstexte so konkretisieren, dass sie für den Empfänger (insbesondere für Ärztinnen und Ärzte) für den individuellen Fall verständlich sind.

Auf der Website des IQTIG stehen Informationen für Endanwender zu den einzelnen Verfahren und zur Erleichterung der Dokumentation bereit. Zu Letzterem gehören die Dokumentationsbögen, Ausfüllhinweise und Anwenderinformationen. Diese Dokumente, die sich an Leistungserbringer richten, die Anwender der QS-Software sind (z. B. Ärztinnen oder Ärzte), sind unter Berücksichtigung verschiedener Anforderungen möglichst anwenderorientiert und verständlich formuliert. Neben der Verständlichkeit werden beispielsweise auch Aspekte wie Einheitlichkeit, technische Umsetzbarkeit und Aufwand bei Verfahrensteilnehmern berücksichtigt.

Ab dem Spezifikationsjahr 2019 beinhaltet die Basisspezifikation Vorgaben zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL). Damit werden mit niedergelassenen Leistungserbringern, ambulant am Krankenhaus erbrachten Eingriffen, Krankenkassen und neuen Datenannahmestellen weitere Beteiligte und Leistungen in die gesetzliche Qualitätssicherung einbezogen.

Die vorliegende Technische Dokumentation richtet sich an alle beteiligten Leistungserbringer und die von ihnen beauftragten Softwarehersteller im QS-Verfahren ambulante Psychotherapie. Das serielle Datenflussmodell gem. DeQS-RL ist auf den Seiten des G-BA veröffentlicht:

https://www.g-ba.de/downloads/39-261-3890/2019-07-18_DeQS-RL_Datenflussmodell.pdf

In der Technischen Dokumentation werden die Prozesse jeweils mit Bezug auf die verschiedenen Leistungserbringer, Datenflüsse und Richtlinien dargestellt.

Die Spezifikation richtet sich insbesondere an die Softwarehersteller, die üblicherweise durch die Leistungserbringer mit der Umsetzung der Spezifikation betraut werden und geeignete Softwareprodukte zur Verfügung stellen. Die Programme müssen die Erfassung aller Daten gemäß den oben genannten Richtlinien und dieser daraus abgeleiteten Spezifikation ermöglichen, diese Daten auf Vollständigkeit und Plausibilität prüfen, im vorgegebenen Format an die datenentgegennehmenden Stellen exportieren und fehlerhafte (von den Datenstellen abgelehnte) Datensätze in geeigneter Weise dem Anwender zur Korrektur und zum erneuten Export vorlegen. Darüber hinaus bietet die Spezifikation den Softwareanbietern Hinweise für die Gestaltung der Benutzeroberfläche und die benutzerfreundliche Plausibilitätsprüfung unmittelbar bei der Dateneingabe.

1.3 Releaseplanung

Um Planungssicherheit zu gewährleisten und angemessen auf Fehler reagieren zu können, werden die Termine zur Veröffentlichung von Spezifikationspaketen (Versionen) eines Spezifikationsjahres und zu Rückmeldefristen in der Technischen Dokumentation angekündigt. Tabelle 1

stellt eine entsprechende Übersicht für das Paket der QS-Basispezifikation 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 Spezifikationsjahres am 30. Juni des Vorjahres veröffentlicht.

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

Frist	Meilenstein	Bereitstellung	Bemerkung
nach Beschluss durch das Plenum	Version V01	Veröffentlichung auf der IQTIG-Webseite (http://www.iqtig.org)	Finale Version für freiwillige und landesbezogene Verfahren und QS-Verfahren gem. DeQS-RL
15.09.	Frist für Fehlerrückmeldungen	E-Mail an verfahrenssupport@iqtig.org oder Nutzung der Kommunikationsplattform	
30.09.	Version V02	Veröffentlichung auf der IQTIG-Webseite (http://www.iqtig.org)	Fehlerkorrekturen
20.10.	Frist für Fehlerrückmeldungen	E-Mail an verfahrenssupport@iqtig.org oder Nutzung der Kommunikationsplattform	
November	Version 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 Jahrdurch das BfArM ²

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 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).

² BfArM: Bundesinstitut für Arzneimittel und Medizinprodukte

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
- 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 (<https://forum.iqtig.org/>) 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.

Die Erstellung der Spezifikation erfolgt nach einer mit den Verfahrensteilnehmern abgestimmten Releaseplanung. Diese ist so konzipiert, dass die Spezifikation vor Beginn des Spezifikationsjahres finalisiert und unterjährig nicht mehr aktualisiert wird. Eine unterjährige Aktualisierung sollte nur in Ausnahmefällen erfolgen. Sie sollte nur dann vorgenommen werden, wenn ein gravierender Fehler den Abschluss eines Großteiles der Bögen verhindert und kein anderer „Workaround“ als Übergangslösung gefunden werden kann.

In unterjährigen Updates können keine schnittstellenrelevanten Änderungen oder strukturelle Änderungen an Spezifikationskomponenten vorgenommen werden, da jederzeit die Rückwärtskompatibilität gewährleistet bleiben muss. Das bedeutet, dass z. B.:

- keine neuen Datenfelder in die Dokumentation aufgenommen werden können,
- Regeln nicht von weich auf hart gesetzt werden können,
- Kann-Felder nicht zu Muss-Feldern umgewandelt werden können.

Zusammengefasst: Es können nur aufweichende Maßnahmen, jedoch keine Verschärfungen, in unterjährigen Aktualisierungen vorgenommen werden. Dies ist von hoher Bedeutung, da ansonsten zwar die mit einem Update adressierten Probleme gelöst werden, an anderer Stelle aber auch neue Probleme im Datenfluss entstehen.

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

verfahrensupport@iqtig.org

www.iqtig.org/

A Prozesse

Im Folgenden werden die einzelnen Prozesse und Unterprozesse der Spezifikation beschrieben. In Tabelle 2 werden diese Prozesse in Abhängigkeit von 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	Krankenhaus stationär	Krankenhaus ambulant	Belegarzt*		Er-mäch-tigter Arzt	Ambu-lant nieder-gelas-sene Arzt-praxen	selek-tivver-trag-lich tätige Arzt-praxen
DAS	LKG	LKG	LKG	KV	KV	KV	DAS-SV
Prozesse							
QS-Dokumentation							
Auslösung	A 1.1	A 1.1	A 1.1	A 1.1	A 1.1	A 1.1	A 1.1
Erfassung	A 1.2	A 1.2	A 1.2	A 1.2	A 1.2	A 1.2	A 1.2
Datenexport	A 1.3	A 1.3	A 1.3	A 1.3	A 1.3	A 1.3	A 1.3
Datenübermitt-lung	A 5.2	A 5.2	A 5.2	A 5.2	A 5.2	A 5.2	A 5.2
Rückprotokol-lierung	A 1.5	A 1.5	A 1.5	A 1.5	A 1.5	A 1.5	A 1.5
Sollstatistik							
Erzeugung	A 0	A 0	-	A 0	A 0	A 0	A 0
Datenübermitt-lung	A 0	A 0	-	-	-	-	-
Rückprotokol-lierung	A 1	A 1	-	-	-	-	-
Risikostatistik							
Erzeugung	A 1	-	-	-	-	-	-

Zielgruppe	Krankenhaus stationär	Krankenhaus ambulant	Belegarzt*		Er- mäch- tigter Arzt	Ambu- lant nieder- gelas- sene Arzt- praxen	selek- tivver- trag- lich tätige Arzt- praxen
DAS Prozesse	LKG	LKG	LKG	KV	KV	KV	DAS- SV
Datenübermitt- lung	A 1	-	-	-	-	-	-
Rückprotokol- lierung	A 1	-	-	-	-	-	-

* Belegärztlich erbrachte Leistungen werden dem KH zugeschlagen, im Rahmen des Verfahrens QS PCI der DeQS-RL dem Belegarzt. Ausnahme ist das Verfahren CHE nach DeQS-RL, in dem die belegärztlich erbrachten Leistungen ebenfalls dem KH zugeschlagen werden.

1 QS-Dokumentation

In diesem Kapitel werden die Prozessschritte sowie die in jedem Prozessschritt benötigten Werkzeuge der QS-Dokumentation in Bezug auf die Auslösung, Erfassung, Verarbeitung und Datenübermittlung der QS-Dokumentation des Verfahrens Ambulante Psychotherapie beschrieben (Abbildung 1).

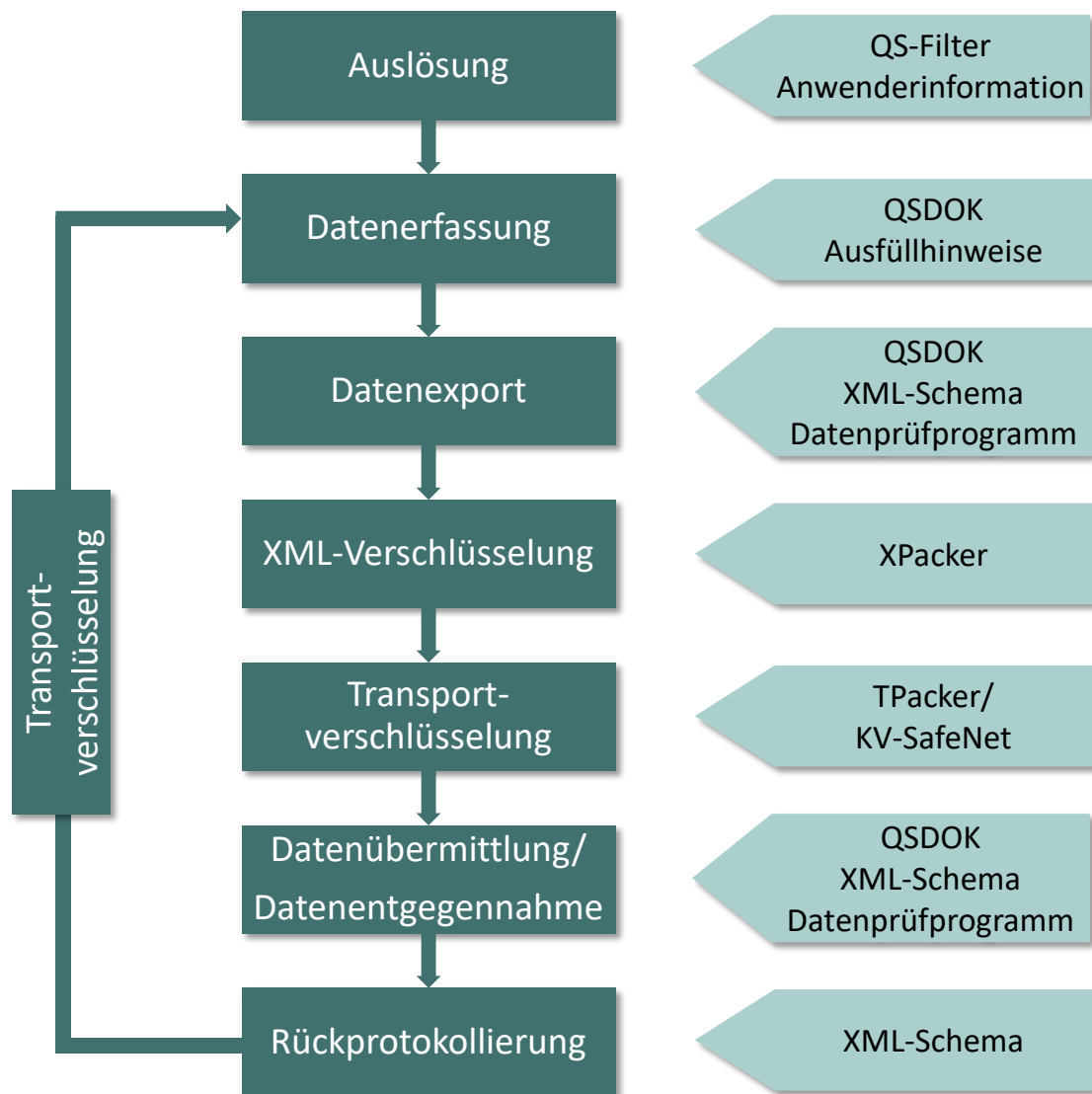


Abbildung 1: Überblick über die Prozesse und Werkzeuge der Datenannahme der QS-Daten

Nachfolgend werden die einzelnen Prozesse und Unterprozesse der QS-Dokumentation beschrieben.

1.1 Auslösung

Die QS-Filter-Software entscheidet für jeden Fall in der Praxis, welche Datensätze (Module) der externen Qualitätssicherung dokumentationspflichtig sind. Mit der vorliegenden Spezifikation können Systeme entwickelt werden, die eine zeit- und prozessnahe Auslösung von QS-Dokumentationen ermöglichen.



Achtung

Der Prozess der Auslösung der Module APSY und PAPSY soll **teilweise automatisch** durch die Softwareimplementierung (QS-Filter-Software, integrierte Software) der Auslösekriterien erfolgen.

Der Algorithmus (Abschnitt B 1.3.2) trifft seine Entscheidung auf der Grundlage der medizinischen Routinedokumentationen (Prozeduren, Gebührenordnungspositionen gemäß EBM-Katalog). Diese sind in jedem Praxisverwaltungssystem (PVS) verfügbar, da sie auch für die Umsetzung der Datenübermittlungsvereinbarung gemäß § 301 Abs. 3 SGB V (kurz: DÜV-301) nötig sind. Die Informationen werden in einzelnen Fällen auch über spezialisierte Systeme, die auf die individuellen Anforderungen im jeweiligen Bereich zugeschnitten sind, zur Auslösung herangezogen. Im QS-Filter-Eingangsdatensatz (Abschnitt B 1.3) ist definiert, welche Daten verwendet werden. Diese basieren auf der KVDT-Datensatzbeschreibung der Kassenärztlichen Bundesvereinigung für den Einsatz von IT-Systemen in der Arztpraxis zum Zwecke der Abrechnung gemäß § 295 Abs. 4 SGB V.

Ein mögliches Szenario für den Einsatz von QS-Filter-Software findet somit an der Schnittstelle zwischen der Anwendungssoftware (PVS/Spezialsystem) und QS-Dokumentationssoftware statt. QS-Filter-Software kann als Service für die anderen Systeme realisiert werden: Über eine Anfrage an den QS-Filter-Service wird dem anfragenden System eine Modulliste geliefert. Bei diesem Szenario ist die QS-Filter-Software zustandsfrei: Sie protokolliert nicht die Erfüllung der Dokumentationspflicht (Soll/Ist). Der Soll-/Ist-Abgleich könnte z. B. direkt in der Anwendungssoftware geschehen. Hierzu müsste die Anwendungssoftware in geeigneter Weise mit der QS-Dokumentationssoftware kommunizieren: Die Anwendungssoftware liefert der QS-Dokumentationssoftware für jeden Behandlungsfall eine Liste der dokumentationspflichtigen Module; die QS-Dokumentationssoftware übermittelt an die Anwendungssoftware den Dokumentationsstatus der Module (in Bearbeitung, abgeschlossen, rückbestätigt).

Dies ist nur eine von vielen Einsatzmöglichkeiten. Am Beispiel wird jedoch deutlich, dass festgelegt werden muss, in welchem System die Statusverwaltung der QS-Dokumentation und ein kontinuierliches internes Vollständigkeitsmonitoring implementiert wird. Die Spezifikationskomponenten zum QS-Filter sind in Kapitel B 1 beschrieben.

Umsetzungsempfehlung

Dem Leistungserbringer sollte ein softwaregestützter unterjähriger Sollabgleich ermöglicht werden, der in regelmäßigen Zeitabständen durchgeführt werden kann.

1.1.1 Der QS-Filter-Eingangsdatensatz

Die QS-Filter-Software bestimmt für jeden Behandlungsfall die dokumentationspflichtigen Datensätze³ (technisch: Module; inhaltlich: Verfahren/Leistungsbereiche). Hierfür soll sie auf bestimmte, bereits dokumentierte Behandlungsdaten des Falles zurückgreifen, die sich im QS-Filter-Eingangsdatensatz wiederfinden (). Während einer Behandlung können Prozeduren und Gebührenordnungspositionen dokumentiert werden, die Teil des QS-Filter-Eingangsdatensatzes sind. Der QS-Filter-Algorithmus basiert zum einen auf diesen medizinischen und zum anderen auf administrativen Datenfeldern.

Der QS-Filter-Eingangsdatensatz enthält somit diejenigen Datenfelder, die in der Datenbank als Codes hinterlegt sind. Diese benötigt der QS-Filter-Algorithmus (Abschnitt B 1.3.2) um ein Ergebnis zu ermitteln, d.h. die für den entsprechenden Fall dokumentationspflichtigen QS-Module. Der Aufbau des QS-Filter-Eingangsdatensatzes wird in Abschnitt B 1.3.1 beschrieben.

1.1.2 Besonderheiten im QS-Verfahren ambulante Psychotherapie

Zur Feststellung der Dokumentationspflicht eines Datensatzes wurde bei der Entwicklung des Verfahrens ambulante Psychotherapie ein QS-Filter definiert, der mit Angaben aus dem PVS automatisiert die Dokumentationspflicht ermitteln sollte. Einige der Filterbedingungen konnten dabei nur durch Dokumentation erfasst werden, da sie im PVS nicht erfasst werden. Es wurden daher sogenannte Verkürzte Bögen entwickelt, die anstelle eines vollständigen Dokumentationsbogens lediglich das Nicht-Vorliegen aller Einschlussmerkmale dokumentierten.

Im ersten Jahr der Erprobung des Verfahrens stellte sich heraus, dass die technischen Voraussetzungen zur Auslösung des Dokumentationsbogens in der Behandlungspraxis nicht immer erfüllt werden. Der letzte Code zur durchgeführten Therapieform und der Pseudo-GOP zur Anzeige des Endes der Richtlinien Therapie lagen nicht immer im selben Abrechnungsquartal. Mutmaßlich wurde ein Teil der dem Sinne des Verfahrens nach zu dokumentierenden Fälle daher nicht als dokumentationspflichtig gekennzeichnet oder bereits getätigte therapiebegleitende Dokumentationen nicht exportiert.

Zur Erarbeitung eines Filters, der auf verkürzte Bögen verzichten kann und der von der Verwendung der Pseudo-GOP zum Anzeigen des Endes der Richtlinien Therapie unabhängig ist, wird ab 2027 auf die vollautomatisierte QS-Auslösung verzichtet. Damit einher gehen neue Modulauslöser und Adminkriteriensowie ein Wegfall der elektronischen Sollstatistik. Die Dokumentationspflicht wird durch die anwendende Person bestimmt. Zusätzlich werden die Bedingungen auf

³ Definiert in der Spezifikationsdatenbank für QS-Dokumentation.

Dokumentationsebene über entsprechende Datenfelder überprüft. (sog. Anwenderbezogene Fallidentifikation).

1.2 Erfassung

Nach Auslösung eines Moduls durch den QS-Filter folgt der Prozess der Erfassung. Hierbei werden Daten entweder automatisch aus dem PVS in die Eingabemaske der Erfassungssoftware übertragen oder durch den Dokumentierenden manuell erfasst.

Neben QS-Daten (Abschnitt A 1.2.3) sind auch einrichtungsidentifizierende Daten wie das Institutionskennzeichen und die QS-relevante Betriebsstättennummer des niedergelassenen Leistungserbringers (Abschnitt A 1.1.1) sowie ggf. patientenidentifizierende Daten (Abschnitt A 0) zu dokumentieren. Die Erfassung ist abgeschlossen, wenn alle zu dokumentierenden Datenfelder unter Berücksichtigung von Abhängigkeiten und Plausibilitätsprüfungen vollständig erfasst wurden (Abschnitt A 1.2.4).

Als Vorlage für die Gestaltung der Eingabemaske (Abschnitt A 1.2.1) durch den Softwareanbieter dient die Datenfelddescription des jeweiligen Moduls (Abschnitt B 2.2).

1.2.1 Gestaltung von Eingabemasken

Die Benutzeroberfläche einer Erfassungssoftware (Graphical User Interface = GUI) soll ergonomisch und anwenderfreundlich gestaltet sein. Gestaltung und Layout der Eingabemaske sind Aufgabe der Softwarehersteller. Neben Anforderungen der Kunden werden üblicherweise firmeninterne Standards bzw. Vorgaben des Betriebssystems (z. B. Windows) für das „look and feel“ berücksichtigt.

Diese Spezifikation definiert als Minimalstandard die für den Anwender sichtbaren Inhalte der Dokumentationsformulare. Als Referenz für die sichtbaren Inhalte dienen die Dokumentationsbögen, die als Bestandteil der Spezifikation durch das IQTIG veröffentlicht werden. Die Dokumentationsbögen werden als PDF-Dokumente bereitgestellt, die aus der Spezifikationsdatenbank automatisch generiert worden sind. Bei den Dokumentationsbögen handelt es sich um Formulare zur Ansicht (Muster), die nicht zur Dokumentation zu verwenden sind.

Tabelle 3 gibt einen Überblick darüber, welche Informationen der Spezifikationsdatenbank (identifiziert durch Tabelle und Attribut) bei der Erstellung der Dokumentationsbögen berücksichtigt werden und somit auch in den Erfassungssystemen sichtbar sein sollen.

Tabelle 3 Informationen aus der Datenbank, welche im GUI verwendet werden

Tabelle	Attribut	Bemerkung	sichtbar für Anwender
Modul	name	Kürzel des Datensatzes (z. B. APSY), erscheint üblicherweise im Titel des Formulars	ja

Tabelle	Attribut	Bemerkung	sichtbar für Anwender
Modul	bezeichnung	Bezeichnung des Datensatzes (z. B. Mammachirurgie), erscheint üblicherweise im Titel des Formulars	ja
Bogen	bezeichnung	Bezeichnung des Teildatensatzes	ja
Bogen-Feld	gliederungAufBogen	Nummer des Datenfelds, dient bei umfangreicheren Bögen zur besseren Orientierung	(ja)
Bogen-Feld	bezeichnung	Bezeichnung des Datenfelds	ja
Bogen-Feld	ergaenzendeBezeichnung	Ergänzende Bezeichnung zum Datenfeld, kann z. B. durch Wahl der Schrift von der Bezeichnung abgesetzt werden	ja
Feld	laenge	Definiert die Länge des Eingabefelds. Für die Gestaltung des Eingabefelds sind weitere Informationen aus der Datenbank wichtig (z. B. <code>Feld.nachKommaLaenge</code>).	ja
Feld	einheit	Einheiten (wie z. B. <code>ml</code>) müssen angezeigt werden.	ja
BasisTyp	format	Formatanweisungen (wie z. B. <code>TT.MM.JJJJ</code>) sollen – sofern nicht durch übergeordnete Eingabefelder unterstützt – für den Anwender angezeigt werden.	(ja)
SchlüsselWert	Code	Bei Schlüsselfeldern sollen die Codes möglichst in Auswahllisten angezeigt werden. Bei einigen Realisierungsvarianten (z. B. Checkbox) kann auf die Anzeige der Codes verzichtet werden.	(ja)
SchlüsselWert	bezeichnung	Bei Schlüsselfeldern müssen die Textdefinitionen der Codes (z. B. in einer Auswahlliste) angezeigt werden. Für die Sortierung sind die Attribute <code>sortierNrVerwendet</code> und <code>zahl</code> der Tabelle <code>Schlüssel</code> relevant (Abschnitt B 2.2.3).	ja
Ab-schnitt	bezeichnung	Die Überschriften sind wichtig für die Strukturierung und das Verständnis des Datensatzes und müssen deshalb in der QS-Dokumentationssoftware angezeigt werden.	ja

zeigt für ein Datenfeld eines Dokumentationsbogens (PDF) den Zusammenhang zu den Informationen, welche in der Datenbank vorhanden sind.



Abbildung 2: Beispiel für Informationen, die in der Oberfläche angezeigt werden sollen (Spezifikation 2027)

Werden Datenfelder (z. B. OPS) eines QS-Datensatzes aus Fremdsystemen über Schnittstellen importiert, so sollen die übernommenen Daten auch in der Erfassungssoftware angezeigt werden. Es ist für den Anwender wichtig, die vollständigen QS-Daten im Kontext eines „QS-Formulars“ zu sehen und auch auf Richtigkeit und Vollständigkeit zu prüfen.

Allgemeine Grundsätze für die Gestaltung der Eingabemaske

Die Grundsätze für die Plausibilitätsprüfungen wirken sich insbesondere auf die Gestaltung der Benutzeroberflächen in Erfassungssoftware aus. Durch die funktionale Gestaltung sollte ein Kompromiss zwischen Dateneingabekomfort einerseits und Zwang zur aktiven Eingabe korrekter Daten andererseits gefunden werden. Im Folgenden werden die Regeln für die Gestaltung von Benutzeroberflächen aufgeführt:⁴

- Keine Suggestion von Feldinhalten durch Vorbelegung (Defaults): Oberstes Prinzip bei der Gestaltung der Benutzeroberflächen ist, dass dem Anwender des Programms keine Angaben suggeriert werden. Insbesondere darf keine Vorbelegung mit Standardwerten erfolgen, die den „Nicht-Problemfall“ dokumentieren.



Hinweis

Eine Vorbelegung mit Standardwerten ist nicht zulässig. Die Übernahme von im PVS vorhandenen Angaben in die QS-Dokumentation ist hingegen zulässig.

Eine Ausnahme besteht für die in Tabelle `Field` der QS-Dokumentationsdatenbank definierten Strukturparameter, die durch den Softwareanbieter im Bogen vorbelegt werden dürfen.

Beispielsweise können die Angaben der Datenfelder `IKNRKH`, `STANDORT`, `STANDORTAUFN`, `STANDORTICD`, `STANDORTOPS`, `BSNR`, `FACHABT`, `BSNRAMBULANT`, `NBSNRAMBULANT` und `VERsstufe` als Strukturparameter vorbelegt werden, wenn diese Angaben vorliegen.

⁴ Externe Systeme, die Daten an ein Erfassungsprogramm übergeben, sollten diese Grundsätze sinngemäß anwenden.

Die Frage, ob z. B. während der Behandlung Komplikationen aufgetreten sind, darf hingegen nicht mit einem Standardwert „nein“ vorbelegt werden, wenn hierzu keine Angaben im PVS enthalten sind.

- Verwendung der vorgeschriebenen Fehler- und Warnmeldungen bei feldübergreifenden Regeln: Die Fehler- und Warnmeldungen sind so formuliert, dass sie möglichst nicht suggerieren, auf welche Weise widersprechende Angaben korrigiert werden sollen. Insofern sollen sie wörtlich übernommen werden.
- Keine zusätzlichen Ober-/Untergrenzen für Maße, Zeitdauern und Anzahlen: Außer den durch die Datenfeldbeschreibung und die Plausibilitätsregeln vorgegebenen Wertebereichen darf in Erfassungsprogrammen keine Einengung möglicher Merkmalsausprägungen in Wertefeldern erfolgen.
- Zwang zur aktiven Entscheidung zwischen „ja“ und „nein“: An entsprechenden Stellen in den Dokumentationsbögen, bei denen die Auswahl „0“ (nein) und „1“ (ja) (vgl. z. B. Schlüssel JN) zu treffen ist, darf keine Voreinstellung des Wertes im Eingabefeld erfolgen. Es besteht somit der Zwang zur Eingabe eines Wertes. Nur an Stellen, an denen im Erfassungsformular lediglich „1“ als Option angegeben wird, soll die Nicht-Eingabe eines Wertes als Verneinung interpretiert werden. Hintergrund dieser Differenzierung ist, dass einerseits in qualitätskritischen Bereichen eine Unterscheidung zwischen „keine Angabe“ und „nein“ erfolgen muss, es andererseits der Benutzerakzeptanz abträglich ist, wenn diese Systematik auch an allen anderen Stellen durchgängig verfolgt wird.
- Kein automatisches Verändern von Feldinhalten in Abhängigkeit von anderen Feldinhalten.

Beispiel:

Wenn im Feld „Wurde ein, dem individuellen Behandlungsfall angemessenes, psychodiagnostisches Testverfahren durchgeführt?“ zunächst „ja“ angegeben und daraufhin das folgende Datenfeld „Wurde ein standardisiertes Testverfahren durchgeführt?“ mit ja beantwortet wurde, und dann im ersten Feld „ja“ zu „nein“ geändert wird, darf diese Änderung nicht automatisch zur Entfernung des Kreuzes bei der zweiten Frage führen. Vielmehr soll eine Fehlermeldung erfolgen und der Anwender gezwungen sein, zunächst das Kreuz im ersten Datenfeld zu entfernen, bevor das zweite verneint werden kann.

Umrechnung von Einheiten bei numerischen Feldern

In Einzelfällen ist es aus Anwendersicht hilfreich, wenn die Eingabemaske die Eingabe von Messwerten in Einheiten ermöglicht, die von den spezifizierten Einheiten abweichen.

Gestaltung von Eingabemasken mit Layout-Feldgruppen

Sogenannte Layout-Feldgruppen sollen in der Erfassungsmaske separat kenntlich gemacht werden. Hierbei handelt es sich um Datenfelder, die zu einer logischen Gruppe zusammengefasst werden können. Feldgruppen können Filterfelder oder abhängige Felder beinhalten. Abhängige

Felder von Layout-Feldgruppen werden auf den generierten Dokumentationsbögen eingegraut dargestellt. Hierbei handelt es sich um Feldgruppen, bei denen das Attribut `grauWennNegativ` in der Datenbanktabelle `FeldGruppe` gesetzt ist. Diese sollen für die Gestaltung von Eingabemasken verwendet werden.

Die Eigenschaften von Layout-Feldgruppen sind in Abschnitt B 2.3.6 erläutert.

Empfehlung zur Umsetzung von Layout-Feldgruppen

Ist das Attribut `grauWennNegativ` gesetzt, so darf die Benutzereingabe für die abhängigen Felder durch die Erfassungssoftware deaktiviert werden, falls die negative Filterbedingung zutrifft. Bei der Umsetzung muss Folgendes sichergestellt werden:

- Nach jeder Änderung der Inhalte der Filterfelder im Erfassungsformular muss das Programm die Filterbedingung der Feldgruppe evaluieren und ggf. eine Aktualisierung der Oberfläche durchführen.
- Die Benutzereingabe für die abhängigen Felder darf nur dann deaktiviert werden, wenn keines dieser Felder ausgefüllt ist. Ansonsten ist der Anwender auf eine Plausibilitätsverletzung hinzuweisen.
- Wenn nach einer Benutzereingabe die positive Filterbedingung zutrifft, so sind ggf. vorher deaktivierte Eingabefelder wieder zu aktivieren.
- Deaktivierte Felder dürfen nicht ausgeblendet werden.

1.2.2 Einrichtungsideifizierende Daten

In Hinblick auf eine einrichtungs- bzw. standortbezogene Auswertung und Berichterstattung sind einrichtungsideifizierende Daten in der QS-Dokumentation zu dokumentieren.

Aktualisierung, Plausibilisierung und Mapping von Einrichtungsideifizierenden Daten

In der Praxis kann es zu unterjährigen Wechseln der BSNR oder von Standortangaben kommen, beispielsweise durch Fusionen. Wenn dieser Fall eintritt, müssen Standortangaben und BSNR im Rahmen der QS-Bögen korrigiert werden. Hieraus ergeben sich folgende Anforderungen.

■ Leistungserbringer

Der Leistungserbringer muss einen Standortwechsel oder fehlerhaft übermittelte Standortangaben über ein Update korrigieren. Ein BSNR-Wechsel muss über ein Storno des gesamten Datenbestandes und Übermittlung der Datensätze mit der neuen BSNR erfolgen.

Standorttrennung in mehrere Standorte

Da ein Standort auch in mehrere Standorte getrennt werden kann, sind folgende Vorgaben zu beachten.

- Wenn der Leistungserbringer alle oder zumindest einzelne Module einem Standort zuordnen kann, ist ein nachträgliches Standort-Mapping möglich und muss durch den Leistungserbringer durchgeführt werden.

- Wenn die Zuordnung, z.B. auch in Bezug auf Behandlung und Entlassung nicht eindeutig ist, muss die Korrektur auf Datensatzebene und damit durch Unterstützung der Software erfolgen. Hierbei muss der Leistungserbringer entscheiden, welchem Teil die Dokumentation zugeschrieben werden soll.
- Wenn keine eindeutige Zuordnung auf die getrennten Standorte möglich ist, muss im Einzelfall geklärt werden, ob eine gemeinsame Auswertung als Einheit möglich ist.

Prüfung von einrichtungsidentifizierenden Daten

Aufgrund der Bedeutung des LE-Pseudonyms für das Erstellen von Auswertungen sind korrekte, vollständige und plausible Angaben von besonderer Wichtigkeit. Aus diesem Grund erfolgt zusätzlich zu den Prüfungen durch die zuständigen Datenannahmestellen eine Standortprüfung im Sinne eines Warning bei Datenentgegennahme bei der Bundesauswertungsstelle.

Einrichtungsidentifizierende Daten der Arztpraxis bzw. des niedergelassenen Leistungserbringers

Die wesentliche Angabe zur Identifikation eines ambulanten Leistungserbringers ist die von der von der Kassenärztlichen Vereinigung (KV) vergebene Betriebsstättennummer (BSNR; Feld BSNRAMBULANT). Sie identifiziert die Praxis als abrechnende Einheit. Dabei umfasst der Begriff Praxis auch Psychotherapeutische Praxen, Medizinische Versorgungszentren (MVZ) und Institute. Dokumentation der Felder KASSEIKNR, PERSONENKREIS und VERSICHERTENIDNEU

Die EDV-Systeme der Praxen sollen die oben genannten Felder, die in der Regel über die elektronische Gesundheitskarte (eGK) eingelesen werden, automatisiert in die QS-Dokumentationssoftware übernehmen. Zur Erfassung der Felder ist statt des direkten Auslesens der Karte auch die elektronische Übernahme aus dem PVS möglich. Praxen müssen in Zusammenarbeit mit Softwareanbietern geeignete Wege finden, die Daten für die Felder in den QS-Dokumentationsbögen der PID-Module verfügbar zu machen.

Die fehlerfreie Übertragung der Versichertendaten in die QS-Software ist eine zentrale Voraussetzung für die leistungserbringer-, einrichtungs- und sektorenübergreifende Qualitätssicherung, daher ist eine manuelle Erfassung der PID-Daten unmittelbar im QS-Dokumentationsbogen durch den Anwender nicht zulässig.



Achtung

Dokumentation der Felder KASSEIKNR, PERSONENKREIS und VERSICHERTENIDNEU

Liegen Daten zur elektronischen Gesundheitskarte (eGK) zum Zeitpunkt der Erfassung noch nicht vor, erlischt **nicht** die Dokumentationspflicht der Felder KASSEIKNR, PERSONENKREIS und VERSICHERTENIDNEU für GKV-Versicherte. Die Angabe ist bei Vorliegen der elektronischen Gesundheitskarte zu dokumentieren.

**Dokumentation und Export patientenidentifizierender Daten (VERSICHERTENIDNEU)**

Ergänzend zum PID-Feld `VERSICHERTENIDNEU` werden das Institutionskennzeichen der Krankenkasse der Versichertenkarte (`KASSEIKNR`) und der besondere Personenkreis (`PERSONENKREIS`) erfasst. Diese Felder stehen in enger Verknüpfung mit dem PID-Feld `VERSICHERTENIDNEU`. Sie dienen zur Identifikation davon, ob es sich bei dem Behandlungsfall um einen GKV-Versicherten handelt und ob somit PID-Daten zu erfassen sind. Gleichermäßen dient es der Sicherstellung, dass keine PID-Daten von Nicht-GKV-Versicherten exportiert werden.

Das Ersatzfeld `versichertenidgkv` beinhaltet die eGK-Versichertennummer, wenn diese dem vorgegebenen Format entspricht und das 9-stellige Institutionskennzeichen der Krankenkasse der Versichertenkarte mit der Zeichenkette '10' beginnt und kein besonderer Personenkreis vorliegt. Ist diese Bedingung nicht erfüllt, ist das Ersatzfeld leer. Dies gilt auch für den Fall, dass das Institutionskennzeichen der Krankenkasse nicht 9-stellig vorliegt.

Das Ersatzfeld `kasseiknr2Stellen` beinhaltet die ersten beiden Zeichen des Institutionskennzeichens der Krankenkasse der Versichertenkarte, wenn dieses Feld 9-stellig und ausgefüllt ist. Ist dies nicht der Fall, ist das Ersatzfeld leer. Dieses Ersatzfeld wird zweifach exportiert: Innerhalb der QS-Daten sowie im vorgesehenen Datencontainer der Patientendaten. Damit ist die Information für alle Datenannahmestellen, die Vertrauensstelle und die Bundesauswertungsstelle einsehbar und eine Prüfung, dass nur für GKV-Versicherte PID-Daten bzw. ein Patientenpseudonym vorliegt, möglich. Im Minimaldatensatz wird das Ersatzfeld `kasseiknr2Stellenmds` exportiert.

Das Ersatzfeld `versichertenstatusgkv` beinhaltet den Wert 1, wenn das 9-stellige Institutionskennzeichen der Krankenkasse mit der Zeichenkette '10' beginnt und kein besonderer Personenkreis vorliegt. Ist dies nicht der Fall, beinhaltet das Ersatzfeld den Wert 0. Im Minimaldatensatz wird das Ersatzfeld `vstatusgkvmds` exportiert.

**Achtung
Datenexport von PID-Modulen**

Das Modul ambulante Psychotherapie (APSY und PAPSY) wird ausschließlich bei in der gesetzlichen Krankenversicherung versicherten Patienten ausgelöst. Damit werden für diese Patienten weder PID noch QS-Daten erfasst oder exportiert.

Um die Stabilität und die Sicherheit des Verfahrens bei allen Beteiligten zu gewährleisten, werden die zur Identifikation von GKV-Versicherten notwendigen Datenfelder (z. B. das Feld

KASSEIKNR und das zugehörige Ersatzfeld kasseiknr2Stellen) mit allen bestehenden Regeln und Prüfalgorithmen auch für diejenigen QS-Verfahren wie ambulante Psychotherapie beibehalten, die ausschließlich GKV-versicherte Patienten umfassen (ein Ausschluss der Nicht-GKV-Versicherten erfolgt dort bereits über den QS-Filter-Algorithmus).

Bestätigung oder Korrektur fehlender oder fehlerhafter patientenidentifizierender Daten (PID) zur Follow-up-Erhebung

Werden fehlerhafte Daten zum „Institutionskennzeichen der Krankenkasse der Versichertenkarte“ und zur „eGK-Versichertennummer“ in die QS-Dokumentationssoftware übertragen, können die Datenfelder nicht korrekt oder nur unvollständig dokumentiert werden. Eine manuelle Korrektur fehlerhafter Angaben der beiden Datenfelder ist unzulässig.⁵

Liegen in der QS-Dokumentationssoftware fehlerhafte Angaben zum „Institutionskennzeichen der Krankenkasse der Versichertenkarte“ und zur „eGK-Versichertennummer“ vor, sollte die interne EDV-Abteilung oder der entsprechende Softwareanbieter frühzeitig kontaktiert werden, um diese Werte zu korrigieren und um den Fall korrekt abzuschließen.

Dokumentation der Adressdaten in Modulen zur Patientenbefragung (z.B. PAPSYP)

Die EDV-Systeme der Krankenhäuser und der Arztpraxen sollen die Adressdatenfelder in Modulen zur Patientenbefragung (z.B. PAPSYP) analog der Regelung zur Angabe der eGK in PID-Verfahren, automatisiert in die QS-Dokumentationssoftware übernehmen. Die Angaben werden über die elektronische Gesundheitskarte (eGK) eingelesen und wurden daher in der QS-Basispezifikation für Leistungserbringer entsprechend der eGK-Spezifikation definiert. Zur Erfassung der Felder ist statt des direkten Auslesens der Karte auch die elektronische Übernahme aus dem PVS möglich. Praxen müssen in Zusammenarbeit mit Softwareanbietern geeignete Wege finden, die Daten für die Felder in den QS-Dokumentationsbögen verfügbar zu machen.

Bei den als Muss-Felder spezifizierten Datenfeldern ist sicherzustellen, dass die Datenfelder entsprechend den Regelungen aus Abschnitt B 1.4.6 Fehlerprüfung nicht leer gelassen werden dürfen.

Tabelle 4: Adressdaten im Modul PAPSYP

zeileAuf-Bogen	bezeichnung	ergaenzendeBezeichnung	fkFeld	fkBogen	fkMussKann
23	Nachname	Schreibweise entsprechend den Vorgaben der eGK	NNAME	PAPSYP:B	M
24	Vorname	Schreibweise entsprechend den Vorgaben der eGK	VNAME	PAPSYP:B	M

25	Adresszusatz	Schreibweise entsprechend den Vorgaben der eGK	ADZUSATZ	PAPSY:B	K
15	Straße und Hausnummer	Schreibweise entsprechend den Vorgaben der eGK	STRASSE	PAPSY:B	M
16	Postleitzahl	Schreibweise entsprechend den Vorgaben der eGK	PLZ10	PAPSY:B	M
17	Wohnort	Schreibweise entsprechend den Vorgaben der eGK	ORT	PAPSY:B	M

Bei der Übernahme der Angaben von der eGK oder aus dem PVS ist sicherzustellen, dass die Angaben unverändert in die QS-Dokumentation übernommen werden. Beispielsweise ist das Auffüllen von leeren Stellen mit Füllzeichen (z.B. Nullen) nicht zulässig.

1.2.3 QS-Daten

Es sind alle erforderlichen Daten zu dokumentieren. Hierbei kann eine automatische Übertragung der QS-Daten aus dem PVS möglich sein.

QS-Daten können Abrechnungskodes beinhalten, beispielsweise Diagnosen in Form von ICD-Kodes. Grundsätzlich ist hier eine Übereinstimmung aller Kodes (z. B. Diagnosen) mit den Abrechnungsdaten des Behandlungsfalls sicherzustellen. Alle Entlassungsdiagnosen, die in inhaltlichem Zusammenhang mit der in dem jeweiligen QS-Verfahren dokumentierten Leistung stehen, sind daher in das Datenfeld „Entlassungsdiagnose(n)“ [ENTLDDIAG] zu übertragen.

Der Softwarehersteller muss (z.B. bei einer prozessbegleitenden Auslösung oder anderen Änderungen im Nachhinein) sicherstellen, dass Aktualisierungen im PVS zwischen der Anwendungssoftware und der QS-Dokumentationssoftware kommuniziert werden.

Grundsätzlich ist hier eine Übereinstimmung aller relevanten Angaben (z. B. Diagnosen) am Ende des Behandlungsfalls sicherzustellen.

Beispiele:

- Das Datenfeld zu Diagnosen wurde dokumentiert und mit den zu Beginn der Dokumentation vorliegenden ICD-Kodes befüllt. Im Nachgang wurden im KIS jedoch weitere ICD-Kodes kodiert. Der Dokumentationsbogen muss entsprechend (ggf. im Sinne eines Updates) aktualisiert werden.

Dies ist beispielsweise für die Berechnung von Indikatoren oder im Rahmen der Risikoadjustierung von Relevanz.

- Der Bogen eines bestimmten Moduls wurde ausgelöst. Im Nachgang wurde jedoch im PVS eine Ausschlussdiagnose ergänzt. Der Bogen muss storniert werden.

Dies ist beispielsweise für das Erreichen einer Dokumentationsrate von 100 % von Relevanz.

- Zwei bereits ausgelöste und versendete Bögen werden nach einer nachträglichen Fallzusammenführung zu einem Bogen zusammengefasst. Der Softwareanbieter muss die Zusammenführung der beiden Dokumentationsbögen sicherstellen (z.B. durch Stornierung eines Bogens und Update des anderen Bogens).

Dies ist beispielsweise für das Erreichen einer Dokumentationsrate von 100 % und zur korrekten Ermittlung von Fallzahlen von Relevanz.



Achtung

Daten im QS-Bogen müssen regelmäßig aktualisiert werden, um mit dem aktuellen Stand der Daten im PVS übereinzustimmen

Eine automatisierte Aktualisierung von Datenfeldern im Dokumentationsbogen kann ein Softwareanbieter nur durchführen, wenn es sich um Daten handelt, die im PVS vorliegen. Hierbei handelt es sich in der Regel um Daten, die auch im Rahmen der technischen Anlagen nach § 301 und § 21 definiert sind. Die Verpflichtung der automatisierten Aktualisierung besteht für Softwareanbieter daher nur für die im **QS-Filter-Algorithmus genutzten Felder** des jeweiligen Moduls (filterrelevante Datenfelder) (Abschnitt B 1.3).

Die Aktualisierung manuell zu befüllender Datenfelder liegt in der Verantwortung des Leistungserbringers.

Besonderheiten bei ICD-Kodes

Im ambulanten Sektor muss ein Kennzeichen zur Diagnosesicherheit angegeben werden. Das Zusatzkennzeichen zur Diagnosesicherheit ist in den ICD-Listen der Spezifikationsdatenbanken nicht enthalten. Das Zusatzkennzeichen ist für ambulante Fälle dennoch zu dokumentieren.

1.2.4 Plausibilitätsprüfungen

Fehlende und widersprüchliche Angaben in den Datensätzen sollen durch umfangreiche Plausibilitätsprüfungen verhindert werden. In der QS-Dokumentationssoftware muss die vollständige Plausibilitätsprüfung für jeden Datensatz spätestens bei Dokumentationsabschluss erfolgen. Teile der Plausibilitätsprüfungen sollen bereits während der Erfassung erfolgen. Dadurch wird sichergestellt, dass ein aufwendiges Korrekturverfahren – Verschlüsselung und Übermittlung der Datensätze, Prüfung durch DAS, VST oder BAS, Fehlerprotokollierung über die DAS (Datenflussprotokoll), Korrektur der Dokumentation und erneute Übermittlung des Datensatzes – weitgehend entfällt.

Die Datenannahmestellen führen für jeden Datensatz alle harten Plausibilitätsprüfungen der Spezifikation durch. Bei einer Regelverletzung ist der Datensatz zurückzuweisen. Die Datenannahmestellen dürfen keine zusätzlichen (in der Spezifikation nicht definierten) Plausibilitätsprüfungen durchführen.

Es gelten folgende Grundsätze für die Plausibilitätsprüfung:

- Alle Felder müssen ausgefüllt sein, wenn andere logische Sachverhalte dem nicht entgegenstehen.
- Jedes Feld, das auszufüllen ist, muss einen sinnvollen Feldinhalt haben.
- Es wird jede harte Plausibilitätsprüfung vorgenommen, die definiert ist.
- Harte Plausibilitätsprüfungen werden nur vorgenommen, wenn Sachverhalte zwingend miteinander gekoppelt sind.
- Es werden keine Sachverhalte suggeriert (keine Default-Werte, keine Vorbelegungen, keine Profile. Fehlermeldungen werden vorgegeben).
- Keine Angabe (bzw. kein Feldinhalt) wird ergänzt oder gelöscht.

Arten der Plausibilitätsprüfungen

Es wird zwischen drei Arten von Plausibilitätsprüfungen unterschieden:⁶

- harte Prüfungen
- weiche Prüfungen in der QS-Dokumentationssoftware

Die zwei Arten der Prüfungen werden in unterschiedlichen Kontexten (QS-Software bzw. Datenentgegennahme) durchgeführt und haben unterschiedliche Konsequenzen. Tabelle 5 gibt einen Überblick:

Tabelle 5: Arten der Plausibilitätsprüfungen

Art der Prüfung	Kürzel	Prüfung durch QS-Software	Prüfung durch Datenannahmestelle	Konsequenz: Verhindert Dokumentationsabschluss/Datenentgegennahme
Hart	H	Ja	Ja	Ja
Weich	W	Ja	Optional	Nein

Harte Prüfungen

Harte Prüfungen sind sowohl in der QS-Dokumentationssoftware als auch bei der Datenentgegennahme anzuwenden. Bei einer harten Regelverletzung ist:

- ein Dokumentationsabschluss der QS-Dokumentation eines Vorgangs unzulässig.
- ein Datensatz von der entgegennehmenden Stelle zurückzuweisen.

Die in der Technischen Dokumentation und der Datenbank definierten Plausibilitätsprüfungen sind hart, außer wenn sie explizit als weich oder warnend gekennzeichnet sind.

Weiche Plausibilitätsprüfungen

Die weichen Plausibilitätsprüfungen der Spezifikation sind von der QS-Dokumentationssoftware bis spätestens zum Dokumentationsabschluss durchzuführen. Bei einer Regelverletzung erhält

⁶ Die Komponentensicht ist in Abschnitt B 2.4 „Plausibilitätsprüfungen“ beschrieben.

der Benutzer einen Warnhinweis, anhand dessen er entscheidet, ob eine Änderung von Feldinhalten notwendig ist. Ebenso wie harte Plausibilitätsprüfungen müssen weiche Regeln immer vom Softwareanbieter umgesetzt werden.

Einzelregeln

Einzelregeln sind in der Datenbank zur QS-Dokumentation in Regelsyntax in der Tabelle `Regeln` hinterlegt. Außerdem gibt es Regeln, die nur in Form von Feldeigenschaften – nicht aber in Regelsyntax – in der Datenbank hinterlegt sind. Die standardisierten Fehlertexte dieser Prüfungen sind Abschnitt B 2.3.7 zu entnehmen.

Die QS-Dokumentationssoftware muss sowohl die harten als auch die weichen feldbezogenen Prüfungen ausführen. Die Evaluation soll direkt bei der Dateneingabe geschehen. Fehleingaben sollen dem Benutzer direkt mitgeteilt werden. Einige Prüfungen erübrigen sich durch adäquate Gestaltung von Eingabemasken, z. B. durch Bereitstellung von Auswahlmenüs für Schlüsselcodes. Bei Regelverletzung muss die QS-Dokumentationssoftware dem Benutzer verständliche Fehlertexte anzeigen.

Bei der Datenentgegennahme sind alle harten Prüfungen zu evaluieren und bei Regelverletzung die unten definierten standardisierten Fehlertexte im Rahmen des im Abschnitt B 2.3.7 beschriebenen Korrekturverfahrens an die Einrichtung zu übermitteln.

Feldgruppenregeln

Datenfelder (Bogenfelder) eines Moduls können zu einer Feldgruppe zusammengefasst werden, um logische Abhängigkeiten von Bogenfeldern abzubilden. Das bedeutet in der Praxis, dass der Anwender daran gehindert wird, Felder mit Werten auszufüllen, die der Logik der Feldgruppe widersprechen.

Die explizite Definition von Feldgruppen strukturiert sowohl die Bogenfelder als auch die Plausibilitätsregeln, indem diese die Bogenfelder eines Moduls zu einer logisch zusammenhängenden Gruppe von Feldern zusammenfassen. Die Feldgruppen ergeben sich dabei indirekt aus der Definition von Plausibilitätsregeln.

Die Abbildung von Feldgruppenregeln in der Datenbank für QS-Dokumentation ist in Abschnitt B 2.3.6 erläutert.

1.3 Export der Daten aus der QS-Dokumentation

Die Exportdateien werden beim Leistungserbringer erstellt und dann an die zuständige Datenannahmestelle (DAS) weitergeleitet. Datenannahmestellen nehmen die Datenlieferungen im Rahmen der DeQS-RL entgegen (Abschnitt A 5.1.1). Der Export und die Übermittlung der QS-Daten finden im XML-Format statt.

1.3.1 Erzeugen der Exportdatei

Die zu exportierenden Daten der QS-Dokumentation werden vom Dokumentationssystem in Exportdateien geschrieben und die entsprechenden Vorgänge (jeweils identifiziert durch eine VorgangsnrGuid) im absendenden Dokumentationssystem als „exportiert“ markiert.

Alle vorliegenden Datensätze eines Exportverfahrens sollen gemeinsam in einer Datei und nicht einzeln exportiert werden. Dabei darf eine Exportdatei die Größe von 20 MB und die Anzahl von 10.000 Datensätzen (Fällen) nicht überschreiten. Sollte diese Grenze überschritten werden, müssen die Datensätze (Fälle) durch die Software auf mehrere Exportdateien verteilt werden. Diese Vorgabe bezieht sich auf die entpackte Exportdatei.

Inhalte der Exportdatei

Der Inhalt der einzelnen Exportdateien orientiert sich an den in der „Übersicht über die Exportverfahren“ als zusammen exportierbar gekennzeichneten Modulen (Abschnitt A 5.1.1). Nur Datensätze aus dort entsprechend gekennzeichneten Modulen dürfen gemeinsam exportiert werden.



Hinweis

Dateien, in denen diese Trennung nicht eingehalten wird, werden von der Datenannahmestelle zurückgewiesen.

Export von Teildatensätzen

Beim Export einer QS-Dokumentation durch ein Dokumentationssystem werden die Inhalte der für den betreffenden Vorgang angelegten Teildatensätze hierarchisch in den XML-Code des passenden Basisbogens geschrieben und können nur gemeinsam mit dem Inhalt des Basisbogens exportiert werden. Die Struktur der Einbettung ist durch den Datentyp des Exportmoduls im Schema definiert.

Die Reihenfolge der Unterbögen (Teildatensätze) ist in der Spezifikationsdatenbank festgelegt. In der Tabelle *Bogen* ist in der Spalte *SortierNr* die Reihenfolge der Unterbögen bei der Erfassung und beim Export aufgeführt.

Anonymisierung

Für die Übermittlung der Qualitätssicherungsdaten an die jeweilige Datenannahmestelle müssen personenbezogene Daten durch die QS-Dokumentationssoftware anonymisiert werden. Die Anonymisierung betrifft die Kalendertagesdaten und wird in der Spezifikationsdatenbank (QSDOK) definiert.

Die Kalendertagesdaten werden z. B. in die Differenz in Tagen zwischen dem Kalendertagesdatum und OP-Datum umgerechnet. Die Differenzen sind positiv anzugeben. Ob der Zeitraum prä- oder postoperativ liegt, ergibt sich aus der Bezeichnung des Zeitraums. Für das Entlassungsdatum wird zusätzlich das Quartal berechnet (z. B. „2/2018“ für das zweite Quartal des Jahres 2018). Die Vorschriften zur Anonymisierung von Bogenfeldern sind in den Tabellen *ErsatzFeld* und

ErsatzFuerFeld in der Spezifikationsdatenbank für QS-Dokumentation zu finden (Abschnitt B 0).

Aufbau der Exportdatei

Die innere Struktur der Exportdatei ergibt sich unter anderem aus der Datenfeldbeschreibung der einzelnen Module und den oben beschriebenen Anonymisierungsmaßnahmen zur Erfüllung datenschutzrechtlicher Vorgaben. Aufbauend auf dieser Beschreibung wird ein XML-Schema abgeleitet. Die Struktur der Exportdatei wird durch entsprechende XML-Schemata festgelegt (Abschnitt B 4).

Das XML-Schema beschreibt und definiert die Struktur des XML-Dokuments (Exportdatei) sowie den Inhaltstyp (Datentypen der einzelnen Bögen und Felder).

Die Exportdateien sind wie folgt aufgebaut:

- Header-Bereich enthält die Metadaten (administrative Daten)
- Body-Bereich enthält die tatsächlichen Daten der Datenlieferung



Hinweis

Es ist zu beachten, dass die Zeichenkodierung deklariert wird und diese der UTF-8-Kodierung entsprechen muss.

Sonderzeichen in XML

Das Und-Zeichen (&) und die spitzen Klammern (<, >) müssen geschützt werden, falls sie benötigt werden. Dies kann durch „&“ bzw. „<“ geschehen. Die schließende spitze Klammer (>) kann durch die Zeichenkette „>“ dargestellt werden. Um Attributwerten zu erlauben, sowohl das einfache als auch das doppelte Anführungszeichen zu enthalten, kann der Apostroph (') als „'“ und das doppelte Anführungszeichen (") als „"“ dargestellt werden.

Felder der Exportdatei

Einen Überblick über die zu exportierenden Felder eines Moduls liefert die Abfrage `ExportFelderFürEinModul`.⁷

Export von Listenfeldern

Alle Elemente von Listenfeldern werden exportiert, ohne die Nummer des Listenfelds (im Namen des Exportfelds) an den Namen des Listenfelds anzuhängen (zu Listenfeldern siehe Abschnitt B 2.2.3).

⁷ Metadaten des Datencontainers sind nicht Teil der Spezifikationsdatenbank und werden daher nicht durch die Abfrage ausgegeben.

Beispiel:

XML-Darstellung eines Listenfeldes

```
<ENTLDIAG V="P52.4"/>
<ENTLDIAG V="P90"/>
<ENTLDIAG V="Z76.3"/>
<ENTLDIAG V="Z38.1"/>
<ENTLDIAG V="I63.9"/>
```

Zusatzfelder des Datenexports

Zusatzfelder und administrative Felder im Header, die nicht in der Datenfeldbeschreibung (Tabelle `BogenFeld`) eines Moduls enthalten sind, sind von der QS-Dokumentationssoftware zu füllen.⁸

Die Zusatzfelder sind in der Tabelle `ZusatzFeld` definiert. Das übertragene Speicherdatum `Do-kAbschlDat` (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.

Bei der Organisation im XML-Dokument ist weiter die Abfrage `vExportZieleXml` aus den administrativen Objekten zu berücksichtigen (Abschnitt B 2.7.1). Die Informationen in dieser Abfrage schließen aus, dass bestimmte Felder (wie PID-Daten) dem Element `<qs_data>` zugeordnet werden, indem anhand eines XPath-Ausdrucks (`xmlXPath`) auf die richtige Stelle im XML-Dokument (z. B. `<patient>`) verwiesen wird.

In den in Tabelle in Abschnitt A 5.1 unter „indirekte Verfahren (PID)“ aufgeführten Module sind patientenidentifizierende Daten (PID) zu exportieren.

Organisation der Exportfelder im XML-Dokument

Die Exportfelder sind abhängig von der Modul- und Teildatensatzzugehörigkeit des Datensatzes im Element `<qs_data>`, einem Unterelement des Elements `<case>` unterzubringen (Element `qs_data`, Abschnitt B 4.4.4).

Stornierung

Um den Datensatz zu stornieren, muss `<case>/<case_admin><action>` auf „delete“ gesetzt werden. Die Datenannahmestelle wird dadurch veranlasst, den betreffenden Datensatz einschließlich aller Vorversionen und Teildatensätze als „storniert“ zu kennzeichnen. Der Stornovorgang wird in der Datenbestätigung protokolliert.

⁸ Hier gilt demnach nicht der Grundsatz, dass Felder nicht vorbelegt sein dürfen.

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.

Zur Stornierung eines Datensatzes (Vorgang) ist der Export der entsprechenden administrativen Daten `<case>/<case_admin>` zwingend erforderlich. QS-Daten `<case>/<qs_data>`, die ET-Nummer für das TX-Register `<case>/<patient_tx>`, PID `<case>/<patient>` oder `<case>/<patient_ngkv>` sind nicht erneut zu übermitteln.

1.3.2 Datenprüfung

Zusätzlich zur bereits im Rahmen der Erfassung durchzuführenden feldbezogenen und -übergreifenden Plausibilitätsprüfungen kann nun nach dem Datenexport die gesamte Struktur der XML-Datei durch aus der Spezifikationsdatenbank abgeleitete Schemata geprüft werden (Abschnitt B 4.2).

Welches Schema für einen Leistungserbringer vorgesehen ist, zeigt die `readme.csv` in der Spezifikationskomponente XML-Schema.

Die einfachste Lösung für die Prüfung der Datenstruktur und der Plausibilität liegt in der Nutzung eines Datenprüfprogramms, das auf der Basis von XSLT die aus der Spezifikationsdatenbank QSDOK ausgeleiteten Plausibilitätsregeln in dem XML-Dokument prüft und Verstöße entsprechend im XML-Code dokumentiert. Ein solches Datenprüfprogramm setzt einen Standard für die Güte der Daten, der unbedingt einzuhalten ist. Die Datenprüfung muss an der Exportdatei vor der nachfolgend beschriebenen XML-Verschlüsselung vorgenommen werden.

Das Datenprüfprogramm ersetzt jedoch nicht die Verpflichtung der Softwareanbieter, schon bei der Eingabe der Daten eines Datensatzes, d.h. dokumentationsbegleitend, für die Einhaltung der Plausibilitätsregeln zu sorgen.

Details zur Verwendung können der Dokumentation des Datenprüfprogramms entnommen werden (Abschnitt B 5.1).

1.3.3 XML-Verschlüsselung

Nach Fertigstellung des Dokuments und seiner Prüfung mit dem Datenprüfprogramm sind – sofern vorhanden – die Datenbereiche `/patient`, `/patient_tx` und `/qs_data` mit einer XML-Verschlüsselung anhand des öffentlichen Schlüssels des jeweiligen Zieladressaten zu versehen.

Die Verschlüsselung erfolgt asymmetrisch mit einem öffentlichen Schlüssel.

Tabelle 6 zeigt, welche XML-Elemente mit welchen Schlüsseln zu verschlüsseln sind:

Tabelle 6: Asymmetrische Verschlüsselung der XML-Elemente

Bereich	Verfahren	Datenart	XML-Elemente	Öffentlicher Schlüssel der
kollektivvertraglich	Follow-up	QS-Daten	qs_data	DAS (KV)
		PID-Daten	patient	VST-PSN
selektivvertraglich	Follow-up	QS-Daten	qs_data	BAS
		PID-Daten	patient	VST-PSN

Die öffentlichen Schlüssel der Datenannahmestellen (LKG bzw. KV), der Vertrauensstellen und der Bundesauswertungsstelle stehen den Anwendern als Servicedateien zur Spezifikation zum Download zur Verfügung.⁹

Diese Verschlüsselung baut auf dem W3C-XML-Encryption-Standard auf. Das IQTIG stellt ein Verschlüsselungsprogramm bereit, mit dem eine verfahrenskonforme XML-Verschlüsselung nach diesem Standard durchgeführt werden kann. Das Einbinden der Funktionen des Verschlüsselungsprogramms erfolgt u.a. über einen Befehlszeilenaufruf mit Parametern.

Details zur Verwendung können der Dokumentation der Verschlüsselungsprogramme entnommen werden (Abschnitt B 0).

1.3.4 Ausgangsvalidierung

Als letzte Maßnahme vor der Weiterleitung muss das Dokument gegen das Übertragungsschema auf Gültigkeit geprüft werden. Durch diese Prüfung wird sichergestellt, dass die richtigen Bereiche des XML-Codes verschlüsselt sind, und sie schließt aus, dass kritische Daten versehentlich unverschlüsselt den Leistungserbringer verlassen.

Das für einen bestimmten Leistungserbringer geltende Schema kann der gesonderten Beschreibung der Schema-Familie entnommen werden (Abschnitt B 4.2).

Tabelle 7: XML-Schemata zur Prüfung nach der Verschlüsselung

Bereich	Verfahren	Schnittstelle	Schema
selektivvertraglich	Follow-up	interface_LE_DAS	interface_LE_SV
kollektivvertraglich	Follow-up	interface_LE_DAS	interface_LE_KV



Hinweis

⁹ <https://iqtig.org/datenerfassung/servicedateien/>

Die für die Ausgangsvalidierung zu verwendenden Schemata sind dieselben, wie die bei der Eingangsvalidierung der DAS verwendeten Schemata. Bei nicht bestandener Schemaprüfung wird die Annahme des Dokuments verweigert.

1.3.5 Beispiele für Exportdateien

Beispieldateien unterschiedlicher Schnittstellen können der Spezifikationskomponente „XML-Schema“ entnommen werden. Tabelle 8 zeigt, welche Beispiele für welche Schnittstelle relevant sind:

Tabelle 8: XML-Beispiele

Bereich	Verfahren	Schnittstelle	Beispiele
Selektiv- und kollektivvertraglich	Follow-up	LE vor der Verschlüsselung	interface_LE/pid
		LE/DAS nach der Verschlüsselung	interface_LE_DAS/pid

1.4 Datenübermittlung

Im folgenden Abschnitt werden die abschließend zu unternehmenden Arbeitsschritte Dateibenennung und Datenversand beschrieben.

Vor der Datenübermittlung muss die XML-Verschlüsselung erfolgreich durchgeführt worden sein. Ob sie erfolgreich durchgeführt wurde, kann mit den entsprechenden Schemata für die Ausgangskontrolle geprüft werden.

Die Datenlieferfristen sind in der jeweils gültigen Richtlinie festgelegt. Die Datenübermittlung erfolgt quartalsweise bzw. für Module zur Patientenbefragung monatlich.

Für die sektoren- und/oder einrichtungsübergreifenden Verfahren werden die Datenlieferfristen in den Themenspezifischen Bestimmungen der **DeQS-RL** festgelegt.

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

1.4.1 Dateibenennung

Die Daten werden als XML-Datei an die VST weitergeleitet. Die Exportdatei muss nach einem bestimmten Schema benannt werden (Abschnitt A „Benennung der Exportdateien“ in 5.2.1).

Beispiel:

QS-Daten eines LE

47d16341-9e27-4e75-a27e-b791fbbd2dc8_**Q_LE**.xml

1.4.2 Datenversand via E-Mail – Krankenhäuser

Dieser Abschnitt ist für das QS-Verfahren Ambulante Psychotherapie nicht relevant.

1.4.3 Datenversand via gesicherter Schnittstellen – Arztpraxen/MVZ für ambulant-kollektivvertraglich erbrachte Leistungen

In der Regel wird derselbe Übertragungskanal wie für die Übertragung der Abrechnungsdaten, beispielsweise KIM, genutzt. Beide Übertragungskanäle sowie die Uploadportale der Datenannahmestellen ermöglichen eine sichere Ende-zu-Ende-Verschlüsselung sensibler und vertraulicher Patientendaten im Gesundheitswesen. Die aktuelle Spezifikation für die KIM-Anwendung QSPB finden Sie auf den Seiten der kv.digital¹⁰.

Ggf. ist zu beachten, dass KIM 1.0 Dateien mit einer Dateigröße größer als 15 MB nicht unterstützt.



Hinweis

Anders als bei der Datenübermittlung über nicht gesicherte Übertragungswege (E-Mail), muss die Exportdatei nicht mit dem Verschlüsselungsprogramm TPacker transportverschlüsselt werden.

1.4.4 Datenübermittlung für selektivvertraglich erbrachte Leistungen (Arztpraxen/MVZ)

Die Datenübertragung erfolgt durch das Hochladen der XML-Dateien in ein Upload-Portal für registrierte Ärzte. Die Übermittlung der Daten der selektivvertraglich tätigen Arztpraxen an die DAS-SV setzt eine Registrierung bei dieser voraus. Nähere Informationen können der Webseite der DAS-SV¹¹ entnommen werden.



Hinweis

Anders als bei der Datenübermittlung über nicht gesicherte Übertragungswege (E-Mail) muss die Exportdatei nicht mit dem Verschlüsselungsprogramm (TPacker für eine verpackte und symmetrisch verschlüsselte AES-Datei) transportverschlüsselt werden.

¹⁰ Folgender Link und dort unter Anwendung QSPB: <https://partnerportal.kv-telematik.de/display/KDK/KIM-Anwendungen#KIMAnwendungen-SpezifikationenundAuditsalsPDF-Dokumente>

¹¹ <http://www.vertrauensstelle-gba.de/infoSVLE.html>

1.4.5 Zusammenfassung Datenversand

Tabelle 9 fasst die zuvor beschriebenen Datenübertragungswege zusammen:

Tabelle 9: Zuständige Datenannahmestelle

Bereich	Verfahren	Datenannahmestelle	Datenübertragung	Symmetrische Verschlüsselung
Kollektivvertraglich	Follow-up	KV	Abhängig vom Übertragungskanal der KV (z. B. KIM) ¹²	Nein
selektivvertraglich	Follow-up	DAS-SV	Upload-Portal ¹³	

1.5 Rückprotokollierung

Nach erfolgreicher Eingangsverarbeitung durch die DAS erhält der Leistungserbringer von der DAS auf dem Eingangskanal¹⁴ eine Empfangsbestätigung, die den Erhalt der Exportdatei bestätigt. Nach erfolgreicher Prüfung der XML-Datei erhält der Leistungserbringer zudem ein Datenflussprotokoll mit allen Prüfergebnissen der DAS und ggf. der VST und der BAS.



Hinweis

Da die Protokolle dem LE auf dem Eingangskanal von der DAS zur Verfügung gestellt werden, werden die per E-Mail übermittelten Protokolle von der DAS mit demselben Passwort für Transportverschlüsselung der QS-Daten verschlüsselt und nach demselben Schema benannt.

Das Protokoll ist von der Dokumentationssoftware einzulesen und dem Anwender darzustellen. Gegebenenfalls müssen die fehlerhaften Datensätze korrigiert und erneut übermittelt werden.

Die genaue Struktur ist Abschnitt A 5.3 zu entnehmen.

Regelungen für ein Vorgehen bei Verarbeitungsabbrüchen im besonderen Fehlerfall

Diese Regelungen dienen dazu, ein abgestimmtes Vorgehen durch alle am Datenfluss beteiligten Stellen zu etablieren, um die Datenbestände in allen Instanzen zu konsolidieren, wenn es zu unerwarteten Störungen in der Verarbeitung von Datenlieferungen in der DAS, VST oder BAS kommt. Solche Störungen sind selten und stellen einen Ausnahmefall dar.

¹² In der Regel sollen die Daten analog zum Übertragungsweg der Abrechnungsdaten übermittelt werden.

¹³ <http://www.vertrauensstelle-gba.de/infoSVLE.html>

¹⁴ Es können Abweichungen auftreten. Nähere Informationen sind bei der zuständigen KV bzw. DAS-SV einzuholen.

Solche Störungen können in äußerst seltenen Fällen aufgrund einer fehlerhaften automatisierten Verarbeitung von Datenlieferungen in der DAS, VST oder BAS dazu führen, dass fehlerhafte Datenflussprotokolle versandt werden. Dadurch können beim Leistungserbringer unter Umständen mehrere Datenflussprotokolle zu einer GUID eintreffen, die widersprüchliche Einstufungen der Fälle enthalten. Um diesen Umstand an allen am Datenfluss beteiligten Stellen aufzulösen, müssen die involvierten Vorgänge erneut verarbeitet werden.

Das präferierte Vorgehen in dieser Situation ist ein erneuter Versand aller betroffenen Datensätze, die unter der betroffenen GUID gesandt wurden, durch den Leistungserbringer in einer höheren Version. Dieser Versand muss nach Rücksprache mit der Stelle, bei der die fehlerhafte Verarbeitung aufgetreten ist, und unter Einbindung der anderen Stationen im Datenfluss geschehen und wird deshalb telefonisch von der zuständigen DAS beim betroffenen Leistungserbringer initiiert. Für den erneuten Versand ist beim Leistungserbringer ein Export aller betroffenen Vorgangsnummern einschließlich der Erhöhung der zugehörigen Versionsnummer unter einer neuen GUID durchzuführen. Das im Zuge des Exports erstellte XML-Dokument erhält alle Fälle als Update in der jeweils höchsten Versionsnummer, die das Softwaresystem beim Leistungserbringer erstellt, und 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.

Sollte der Leistungserbringer nicht in der Lage sein, die Daten erneut zu exportieren, so ist der alternative Weg zu wählen, bei dem ein durch alle Instanzen bereits verarbeitetes Dokument (einschließlich Rückprotokollierung mittels Datenflussprotokoll bis hin zum Leistungserbringer) – und somit eine bereits verarbeitete GUID – erneut verarbeitet wird. Das genaue Vorgehen ist im Abschnitt A 5.3 beschrieben. Es bedarf ebenfalls einer telefonischen Abstimmung aller Stationen im Datenfluss. Auf Leistungserbringerseite muss die Software in der Lage sein, zu einer GUID mehr als ein Datenflussprotokoll aufzunehmen, um die Einstufung der Daten gemäß dem zweiten Datenflussprotokoll analog zu den anderen Verarbeitungsstellen anzupassen.



1.6 Datenannahme

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

1.6.1 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 1.4).

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

1.6.2 Fristen für die Datenannahme und -weiterleitung

Die DeQS-Richtlinie zum Verfahren 16 ambulante Psychotherapie sieht in § 16 der themenspezifischen Bestimmungen jeweils den 21. April (Quartal 1), 21. Juli (Quartal 2), 21. Oktober (Quartal 3) und 21. Januar (Quartal 4) des nachfolgenden Jahres für Quartalslieferungen vor. Zusätzlich ist eine Korrekturfrist für das gesamte Spezifikationsjahr bis zum 7. Februar des nachfolgenden Jahres festgelegt. Fällt der 7. Februar auf ein Wochenende oder einen Feiertag, gilt der nächste Werktag. Die Fristenregelung bei Feiertagen ergibt sich aus § 193 BGB.

Die BAS hält ihren Datenservice jeweils bis zum 23.03. offen, um die Verarbeitung der fristgerechten Lieferungen zu garantieren. Die Datenübermittlung erfolgt quartalsweise.

Für die Module zur Patientenbefragung gelten gemäß DeQS-RL monatliche Datenlieferfristen. Eine Korrekturfrist zur Datenübermittlung für das gesamte Spezifikationsjahr ist für die Module zur Patientenbefragung nicht vorgesehen.

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

1.7 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 B2.7.3). 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 (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 A1.3.2).

1.7.1 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

Die KV darf gemäß den G-BA-Richtlinien die QS-Daten einsehen. Daher verschlüsseln Praxen als Leistungserbringer die QS-Daten, die an die 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, so dass 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.

Für die Prüfung der QS-Daten werden die Regeln aus der Spezifikationsdatenbank (QSDOK) herangezogen (Abschnitt B 2.3).

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 5.1).



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

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.

1.8 Pseudonymisierung der LE-Daten

Zu den Aufgaben der DAS gehört für Verfahren gemäß DeQS-RL die Pseudonymisierung der leistungserbringeridentifizierenden Daten (LID) vor der Weiterleitung an die BAS. Hier ist die eindeutige Kennzeichnung der Leistungserbringer (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.

Bei der Pseudonymisierung sind folgende richtlinienspezifische Regelungen zu berücksichtigen:

- Gemäß 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 DeQS-RL muss das Pseudonymisierungsprogramm (PSP) verwendet werden (Abschnitt B 5.3). Welche Elemente durch welche DAS zu pseudonymisieren sind, wird in der folgenden Tabelle aufgeführt.

Tabelle 10: Pseudonymisierung der LE-Daten innerhalb des Elements `care_provider` (DeQS-RL)

Datenannahmestelle	Beschreibung
KV	▪ BSNRAMBULANT ist durch <code>pseudonym</code> zu ersetzen und NBSNRAMBULANT ist durch <code>pseudonym_nbsnr</code> zu ersetzen (falls vorliegend). ▪ Für die Pseudonymisierung wird die BSNRAMBULANT¹⁵ verwendet. ▪ Die Pseudonymisierung ist verfahrensspezifisch durchzuführen. ▪ Die BSNRAMBULANT¹⁶ müssen vor der Weiterleitung entfernt werden. ▪ Die Pseudonymisierung erfolgt mittels des systemweit einheitlichen Verfahrens (PSP).
DAS-SV	▪ Für die Pseudonymisierung wird nur die BSNRAMBULANT verwendet. ▪ Die Pseudonymisierung ist verfahrensbezogen durchzuführen. ▪ Die DAS-SV trägt zusätzlich die Registriernummer der Datenannahmestelle, welche ursprünglich das Pseudonym des Leistungserbringers erstellt hat, in das Attribut <code><pseudonym>/@registration</code>. ▪ Die DAS-SV trägt zusätzlich die KV-Region in das Attribut <code><pseudonym>/@kvregion</code>. Die KV-Region ist aus den ersten zwei Stellen der BSNRAMBULANT abzuleiten. ▪ Mittels des systemweit einheitlichen Verfahrens (PSP).



Für eine landesbezogene Auswertung der selektivvertraglichen Leistungen gemäß DeQS-RL ist eine eindeutige Zuordnung zu den Bundesländern erforderlich. Hierfür ist eine von der KBV bereits spezifizierte Kodierung gut geeignet (KV-Region). Der entsprechende Code (Tabelle 11, 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:

¹⁵ Betriebsstättennummer (ambulant)

¹⁶ Betriebsstättennummer (ambulant)

Tabelle 11: Kodierung der Landesgeschäftsstellen der KV

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.

1.9 Weitere Datenverarbeitungen

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

- Aktualisierung der Absenderinformationen

Aktualisierung der Absenderinformationen

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

Eindeutigkeit der Vorgangsnummern

Die Leistungserbringer erzeugen eine in ihrer QS-Software eindeutige Vorgangsnummer, die ihnen die Reidentifikation der Fälle ermöglicht.

2020 wurde zusätzlich ein 36-stelliger pseudozufälliger Globally Unique Identifier (GUID) eingeführt, der die Datensätze bundesweit eindeutig identifiziert ohne Rückschlüsse auf den Leistungserbringer oder Patienten zuzulassen.

Bei der Durchführung von Stellungnahmeverfahren, bei denen die BAS mit dem Leistungserbringer in Kontakt tritt, werden die originären Vorgangsnummern der Leistungserbringer zur Identifizierung der Vorgänge verwendet. Zusätzlich nennt die BAS auch jeweils die GUID, so dass die Leistungserbringer auch diese oder Teile davon zur Identifizierung von Fällen verwenden können. Dies könnte nötig werden, falls bei einem Leistungserbringer, die eigenen Vorgangsnummern nicht eindeutig sind, weil für ein QS-Verfahren mehrere Software-Instanzen verwendet werden.



Hinweise

Die Datenannahmestellen sollen ab 2020 die von den Leistungserbringern erstellten Vorgangsnummern nicht mehr ergänzen oder ändern. 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.

1.10 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.

Die Notwendigkeit zu dieser Verschlüsselung ergibt sich aus Datenschutzgründen für Dokumente, die über die Vertrauensstelle einer Pseudonymisierung der patientenidentifizierenden Daten (PID) zugeführt werden sollen.

1.11 Datenübermittlung

Im folgenden Abschnitt werden die abschließend zu unternehmenden Arbeitsschritte (Ausgangsprüfung und Datenweiterleitung) beschrieben.

1.11.1 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.



1.11.2 Datenweiterleitung

Für die Übermittlung der Daten über nicht gesicherte Übertragungswege (E-Mail) muss die Exportdatei in eine Transaktionsdatei¹⁷ 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.

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 12).

Tabelle 12: Bisher vorgesehene Datenübertragungswege

Daten-absender	Daten-empfänger	Übertragungsweg	Transportverschlüsselung mit dem TPa-cker
KV	VST	Rest-Service	Nicht erforderlich, da SSL-verschlüsselt

¹⁷ Verpackte und symmetrisch verschlüsselte AES-Datei.

Daten-absender	Daten-empfänger	Übertragungsweg	Transportverschlüsselung mit dem TPa-cker
DAS-SV	VST	Rest-Service	Nicht erforderlich, da SSL-verschlüsselt

1.12 Rückprotokollierung durch die Datenannahmestellen

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

1.12.1 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 5.3.2 zu entnehmen.

1.12.2 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.

1.12.3 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.

1.12.4 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 keine Empfangsbestätigung vorliegen, so ist die VST zu informieren. Sobald eine Empfangsbestätigung

vorliegt, ist der Ansprechpartner für ein fehlendes Datenflussprotokoll die Bundesauswertungsstelle.

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

BAS: verfahrenssupport@iqtig.org

Eine wiederholte Datenübermittlung ohne vorherige Abstimmung mit der VST ist zu vermeiden.

Beispieltext einer Empfangsbestätigung zwischen VST und DAS:

Die Dateneinsendung konnte erfolgreich eingelesen und an die Bundesauswertungsstelle weitergeleitet werden. Bei eventuellen Rückfragen zu dem Datenflussprotokoll kontaktieren Sie bitte die Bundesauswertungsstelle.

(XML-Schema: `interface_DAS_VST\response_receipt.xsd`)

1.13 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.
- Das im Datenflussprotokoll enthaltende LE-Pseudonym bei Verfahren gem. DeQS-RL muss vor der Weiterleitung an den LE depseudonymisiert werden (z.B. `IKNRKH`, Standorte oder `BSN-RAMBULANT`).
- 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.

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@igtig.org vereinbart werden.

1.14 Zusammenfassung

Tabelle 13 stellt einen Überblick über die Aufgabenbereiche der Leistungserbringer in Bezug auf die QS-Dokumentation dar.

Zur Vereinfachung der Darstellung werden folgende Abkürzungen verwendet:

- AP-KV (kollektivvertraglich tätige Praxen): umfasst folgende Fälle:
 - ambulante Leistungen nach § 295 SGB V, § 116 SGB V
- AP-SV (selektivvertraglich tätige Praxen): umfasst folgende Fälle:
 - selektivvertragliche Leistungen der Praxen und der medizinischen Versorgungszentren nach §§ 73 b SGB V
 - Fälle nach §§ 140a ff SGB V, wenn diese unter einer BSNR abgerechnet werden

Weitere Informationen zu den Datenflüssen können Abschnitt A 5.1 entnommen werden.

Tabelle 13: Aufgaben der Leistungserbringer in Bezug auf die QS-Dokumentation

Prozesse	Unterprozesse	AP-KV	AP-SV	Anmerkungen
Auslösung		x	x	Automatisch softwaregestützter Prozess durch Kommunikation mit dem Informationssystem. Implementierung der Auslösekriterien
Erfassung		x	x	Softwaregestützte Dokumentation. Implementierung des Spezifikationsregelwerks (QSDOK). Übernahme von Daten aus dem Informationssystem und manuelle Ergänzungen
Export	Export in XML-Format	x	x	Umsetzung der Vorgaben der QSF und der XML-Schemata DeQS-RL: verfahrensspezifisch
	Konforme Benennung der XML-Datei	x	x	
	Datenprüfung vor der Verschlüsselung	x	x	Mittels DPP, XML-Schemata oder eigener Tools
	Verschlüsselung der XML-Elemente qs_data und patient	x	x	Mittels XPacker
	Ausgangskontrolle (Schema)	x	x	Mittels XML-Schema für die Schnittstelle LE-DAS (mit oder ohne DPP)
Datenübermittlung				
	Upload-Portal		x	Die Registrierung bei der Datennahme auf Bundesebene ist erforderlich.
	KV-Schnittstellen	x		Mit der zuständigen KV abzustimmen.
Rückprotokollierung	Erhalt einer Empfangsbestätigung	x	x	Bestätigung des Erhalts der Exportdatei
	Erhalt der Datenflussprotokolle von der DAS	x	x	Ergebnisse über alle Prüfungen durch die DAS und ggf. durch die VST und die BAS
	Einlesen der Protokolle	x	x	Ausgabe der Fehlermeldungen und ggf. Vornahme Korrekturen



Tabelle 14: Aufgaben der Datenannahmestellen

Prozesse	KV	DAS-SV	Anmerkungen/Eigenschaften
Datenannahme	x	x	Gesicherte Datenübertragung; unterschiedliche Übertragungswege möglich, Registrierung (LKG, DAS-SV), Authentifizierung, Beachtung der Fristen
Eingangskontrolle	x	x	Mittels XML-Schema (mit oder ohne DPP)
Prüfung der Admin-Daten	x	x	Mittels DPP, administrative Objekte in der QSDOK-Datenbank
Entschlüsselung der QS-Daten	x		Mittels XPack (qs_data)
Prüfung der QS-Daten	x		Mittels Plausibilitätsregeln der QSDOK-DB, DPP
LE-Pseudonymisierung	x	x	DeQS-RL: verfahrensspezifisch mittels Pseudonymisierungsprogramm
Eintragung der KV-Region		x	Durch die DAS-SV
Anpassung der Absenderinformationen	x	x	Provider-Tag im header
Verschlüsselung der QS-Daten	x		Mittels XPack (qs_data)
Verschlüsselung der LE-Kennungen bzw. deren Pseudonyme	x	x	Mittels XPack (care_provider)
Ausgangskontrolle (Schema)	x	x	Mittels Schemata für die Schnittstelle zwischen DAS/VST
Empfangsbestätigung	x	x	Benachrichtigung über Erhalt und Weiterleitung der Daten
Datenflussprotokoll im Fehlerfall	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	Voraussetzung: Registrierung der LE bzw. ihrer einzelnen Dokumentationssysteme bei der DAS
Erhalt der Datenflussprotokolle von der BAS	x	x	E-Mail-Verfahren

Prozesse	KV	DAS-SV	Anmerkungen/Eigenschaften
Entpacken der Datenflussprotokolle	x	x	Mittels TPacker
Entpseudonymisierung der LE-Pseudonyme	x	x	DeQS-RL: Mittels Pseudonymisierungsprogramm

2 Risikostatistik

Dieser Abschnitt ist für das QS-Verfahren Ambulante Psychotherapie nicht relevant.

3 Sollstatistik

Mit der Sollstatistik soll nach Abschluss eines Spezifikationsjahres unabhängig von den tatsächlich erfolgten Dokumentationen ermittelt werden, ob ein Leistungserbringer nach den definierten Bedingungen dokumentationspflichtig war. Sie bildet damit eine wichtige Grundlage zur Bestimmung der Vollständigkeit und Vollzähligkeit. Die rechtliche Grundlage zur Ermittlung der Sollstatistik wird durch § 15 der Richtlinie zur datengestützten einrichtungsübergreifenden Qualitätssicherung (DeQS-RL) definiert.

Im Verfahren ambulante Psychotherapie wird ab dem Jahr 2027 auf eine elektronische Sollstatistik verzichtet, um im Gegenzug auf die vollständige Abbildbarkeit des QS-Filters mit Abrechnungsdaten verzichten zu können (A 1.1.2). Somit wird die Verantwortung für eine vollständige Erfassung der dokumentationspflichtigen Fälle in die Hände der Leistungserbringer gelegt.

3.1 Konformitätserklärung

Die Richtigkeit der Angaben in den Datenlieferungen zur Sollstatistik an die jeweilige Datenannahmestelle ist durch die Leistungserbringer mittels einer ausgefüllten und unterzeichneten Konformitätserklärung zu bestätigen. Die Konformitätserklärung ist Teil der Spezifikationskomponenten (Vorlagen_Sollstatistik).

Für die Übermittlung der Konformitätserklärung zur methodischen Sollstatistik soll der Leistungserbringer mithilfe seines Softwareherstellers die Anzahl dokumentationspflichtiger Fälle elektronisch an die jeweilige Datenannahmestelle spätestens bis zum Ende der Korrekturfrist übermitteln. Die Korrektheit der Zahl wird mit einer Unterschrift bestätigt.

Es stehen dem Leistungserbringer folgende Möglichkeiten offen:

1. Postalischer Versand der ausgefüllten und unterzeichneten Konformitätserklärung an die zuständige Datenannahmestelle
2. E-Mail Versand der eingescannten, ausgefüllten und unterzeichneten Konformitätserklärung an die Datenannahmestelle unter Nutzung einer fortgeschrittenen elektronischen Signatur

Bei einem Versand per E-Mail **muss** diese mit einer fortgeschrittenen elektronischen Signatur, welche in einem öffentlich zugänglichen Trust Center hinterlegt ist, versehen werden, welche von der Datenannahmestelle verifiziert werden kann.

Prozess zur Nutzung einer fortgeschrittenen elektronischen Signatur im Rahmen der Übermittlung der Konformitätserklärung von den Leistungserbringern an die Datenannahmestellen auf Landesebene

Die DeQS-RL ermöglicht den Leistungserbringern in Teil 1 § 15 Absatz 3 die Übermittlung der Konformitätserklärung mittels Unterschrift auf postalischem Wege oder per E-Mail unter Nutzung einer fortgeschrittenen elektronischen Signatur (FES). In diesem Abschnitt werden die Vorgaben an die elektronische Signatur sowie die notwendigen Prozesse bei den Leistungserbringern und den Datenannahmestellen skizziert.

Die FES stellt sicher, dass die signierten Inhalte vom Unterzeichner stammen und auf dem Transportweg nachträglich nicht verändert wurden. Hierfür werden elektronische Zertifikate bestehend aus einem öffentlichen und privaten Schlüssel benötigt. Die Zertifikate müssen Klasse 1 Zertifikate sein und von einer anerkannten Zertifizierungsstelle (Trust Center) ausgestellt bzw. signiert worden sein. Die Zertifizierungsstelle stellt das Zertifikat erst nach vorheriger Identitätsprüfung der Person oder Organisation aus, so dass eine Rückverfolgbarkeit der Signatur zum Unterzeichner immer ermöglicht wird. Die im Rahmen der Spezifikation zu nutzenden Zertifikate sind S/MIME-Zertifikate. Eine Nutzung von PGP-Zertifikaten ist aufgrund der fehlenden Identitätsprüfung nicht möglich.

Die durch den Leistungserbringer genutzten Zertifizierungsstellen zur Ausstellung der Zertifikate müssen eine Überprüfung dieser in den Datenannahmestellen ermöglichen, z. B. indem diese bereits in den gängigen Betriebssystemen als vertrauenswürdige Zertifizierungsstellen hinterlegt sind. Krankenhäuser nutzen bereits regelhaft Zertifikate der Zertifizierungsstellen ITSG und DKTIG und können diese auch zur Signierung der Konformitätserklärung nutzen. Diese Zertifizierungsstellen sind manuell als vertrauenswürdige im Signaturprüfungsprozess beim Empfänger zu hinterlegen. In der Spezifikation werden keine weitergehenden Vorgaben hinsichtlich der zu nutzenden Zertifizierungsstelle gemacht.

Der Leistungserbringer (bzw. dessen Dienstleister) übermittelt die unterschriebene und eingescannte Konformitätserklärung als Scan oder Foto in den Formaten PDF oder JPG. In der E-Mail ist standardisiert im Betreff anzuzeigen:

Konformitätserklärung für <IKNR>/<Standort> Spezifikationsjahr <JJJJ>

Der E-Mail-Text kann nach Belieben gewählt werden. Die Konformitätserklärung wird als Anhang zu dieser E-Mail übermittelt. Die Dateigröße darf 1MB nicht überschreiten. Vor dem Versand erfolgt die Signatur der E-Mail. Eine separate Signierung der Konformitätserklärung selbst wird nicht durchgeführt.



Datenfeld „Konformitätserklärung fehlt“

Für die Kommunikation zwischen der Datenannahmestelle und der Bundesauswertungsstelle ist das Datenfeld „Konformitätserklärung fehlt“ in der Tabelle `TdsField` eingefügt worden. Dieses Feld soll nicht vom Leistungserbringer oder Softwareanbieter ausgefüllt werden.

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 DeQS-RL, plan. QI-RL) werden Jahres- und Quartalsberichte (Rückmeldeberichte, Zwischenberichte, Länderauswertungen etc.) von der BAS einerseits für die Landesstellen sowie andererseits für die Leistungserbringer erstellt.

Übertragungsweg

Die Abholung der Auslieferungseinheiten erfolgt 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.

Namenskonventionen

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

Berichtsdokumente

```
<Richtlinie>_<Auswertungsmodul>_<Auswertungsjahr[-Quartal]>_<Produkttyp>_<Auslieferungseinheit>_<Erzeugungsdatum>.[Dateityp]
```

Auslieferungseinheiten

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

Tabelle 15: Dateinamensbestandteile von Auslieferungseinheiten und deren beinhaltete Dateien

Dateinamensbestandteil	Ausprägung (exemplarisch)	Anmerkung
Richtlinie	DEQS	Berichte gem. DeQS-RL
Auswertungsmodul/-typ	Quartalsbericht	Falls mehrere Verfahren
	Quartalsbericht-ambPT	Adressat DAS, z. B. Zwischenberichte gem. DeQS-RL
	Jahresbericht-ambPT	Adressat DAS, z. B. Rückmeldeberichte gem. DeQS-RL

Dateinamensbestandteil	Ausprägung (exemplarisch)	Anmerkung
Auswertungsjahr [-Quartal]	2027	Auswertungsjahr
	2027-Q1	Auswertungsjahr mit Quartals- angabe
Produkttyp	LEAW	Adressat: LE LeistungsErbringer-AusWer- tung
	AV	Auffällige Vorgänge
	ERG	ERG ebnislisten
	LEAE	Adressat: LE LeistungsErbringer-Ausliefe- rungs Ein heit
	t1#kv#a123bcdefg4k	Kurzpseudonym der Betriebs- stättennummer eines LE im Zu- ständigkeitsbereich der KV Test- Land
Erzeugungsdatum	2027-09-15	Format YYYY-MM-DD

Dateinamensbestandteile werden durch Underscores/Unterstriche („_“) voneinander getrennt. Innerhalb eines Dateinamensbestandteils erfolgt eine Trennung optionaler Bestandteile durch einen Bindestrich („-“), z. B. 2027-Q2.

Erläuterung Kurzpseudonym (DeQS-RL)

Datensätze, die im Rahmen von QS-Verfahren gem. der DeQS-RL übermittelt werden, enthalten Leistungserbringer-Pseudonyme. Für die Verwendung der Pseudonyme in Berichten und Dateinamen wird eine Kurzversion des Pseudonyms (Kurzpseudonym) während der Datenannahme im IQTIG erstellt. Kurzpseudonyme von Krankenhausstandorten unterscheiden sich von denen der IKNR und BSNR durch die Verwendung des + anstelle der # an der dritten Stelle nach dem Regionalkürzel.

Beispiele Auswertungen für Verfahren gem. DeQS-RL:

- DEQS_Quartalsbericht-PCI_2020-Q1_LEAE_BW#KV#T427YQXYOU0R_2020-09-17.zip
- DEQS_Jahresbericht-PCI_2020_LEAE_BW+LKG#P429YOMYOU0L_2021-10-03.zip
- DEQS_Jahresbericht-PCI_2020_LEAW_BW+LKG#P429YOMYOU0L_2021-10-03.pdf
- DEQS_Jahresbericht-PCI_2020_ERG_BW+LKG#P429YOMYOU0L_2021-10-03.csv
- DEQS_Jahresbericht-PCI_2020_AV_BW#KV#A129YOM0LYOU_2021-10-03.csv
- DEQS_Jahresbericht-PCI_2020_AV_BW#KV#A129YOM0LYOU_2021-10-03.pdf

5 Allgemeine Regelungen zur Datenübermittlung

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

5.1 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 bestehenden QS-Verfahren gibt es sowohl Daten mit als auch ohne PID. In den Verfahren ist auch die Durchführung einer Pseudonymisierung von leistungserbringeridentifizierenden Daten bei der Zuordnung zu beachten, insofern keine Ausnahmeregelung existiert.

5.1.1 Datenfluss der QS-Daten

Das Verfahren Ambulante Psychotherapie ist ein indirektes Verfahren ohne PID (ohne Follow-Up).

„Ohne FU“ meint hier, dass kein Follow-up mittels PID-Verknüpfung stattfindet.

- **Mit „indirekten Verfahren“ (iv) werden Verfahren bezeichnet, in denen** Datensätze zu Leistungsbereichen gem. DeQS-RL vom Leistungserbringer über die auf Landesebene als Datenannahmestellen (DAS) beauftragten Stellen an die Bundesauswertungsstelle (BAS) übermittelt werden.

Die folgende Abbildung gibt einen Überblick über die im gesamten Bereich Basisspezifikation vorhandenen verschiedenen Datenflüsse:

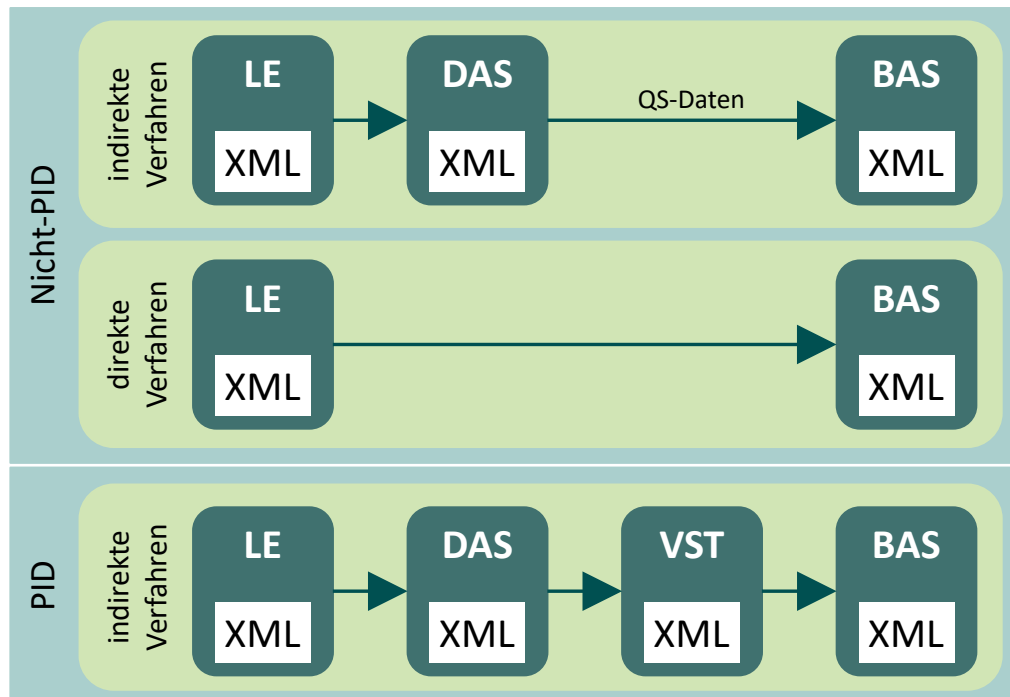


Abbildung 3: Übersicht der Datenflüsse direkte/indirekte PID-/Nicht-PID-Verfahren

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. KIM) 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. Diese Zuordnung kann auch der Abfrage `ExportModuleGemeinsam` der Spezifikation zu Datenserviceinformationen 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 2.7.2). 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 Regelung zur Pseudonymisierung der LE-Daten in der DeQS-RL (verfahrensbezogene Pseudonymisierung), sind die DeQS-Verfahren in getrennten XML-Dateien zu exportieren. So sind Mischlieferungen von Verfahren verschiedener Richtlinien als auch Mischlieferungen von unterschiedlichen DeQS-Verfahren von den Datenannahmestellen abzulehnen. Alle Exportmodule, die in der Abfrage `ExportModuleGemeinsam` der Spezifikation zu Datenserviceinformationen dem gleichen Exportverfahren zugeordnet sind, können gemeinsam exportiert werden.

Beispiele für abzulehnende Datenlieferungen:

- 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 werden in einer Datenlieferung verschickt: Sie sind von der DAS abzulehnen, weil sie nach DeQS-RL einer verfahrensspezifischen und somit unterschiedlichen LE-Pseudonymisierung unterliegen. Eine Aufsplittung dieser Daten in mehr als ein Data-Container innerhalb einer Datei ist nicht zulässig, um die Zusammenführbarkeit unterschiedlicher Pseudonyme in der BAS zu verhindern.
-

5.1.2 Datenfluss der Rückprotokolle

Grundsätzlich sendet der Empfänger einer Datenlieferung dem Absender eine Empfangsbestätigung (Abschnitt A 5.3.2). Die Versendestelle Patientenbefragung und die Bundesauswertungsstelle senden keine Empfangsbestätigung an eine Datenannahmestelle, da die VPB und die BAS das Datenflussprotokoll versenden. Die VST erhält eine Empfangsbestätigung.

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 4). Die Vertrauensstelle (VST-PSN) wird in diesem Datenfluss übersprungen.

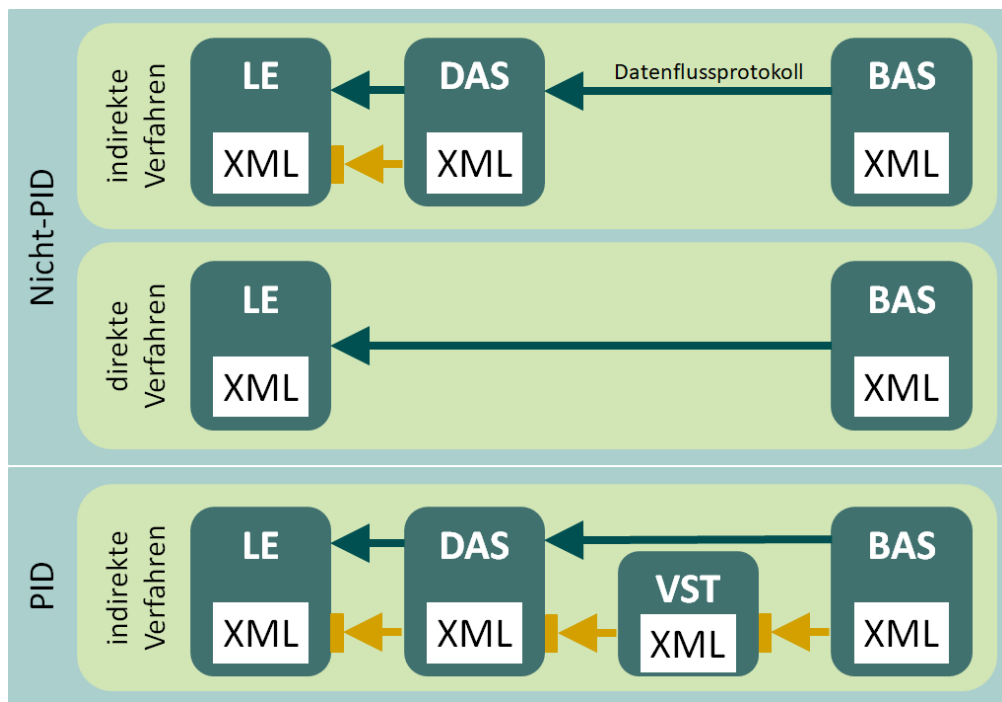


Abbildung 4: Übersicht der Datenflüsse der Rückprotokollierung (orange = Empfangsbestätigung)

5.2 Datenübermittlung

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

5.2.1 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 Übertragungs-
wegen 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 KIM. 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.

Die Adressen der einzelnen DAS werden auf der IQTIG-Homepage aufgeführt. Die dort angegebenen Informationen über die Datenservices werden in einer gesonderten Datenbank bereitgestellt unter: <https://iqtig.org/datenerfassung/spezifikationen/spezifikation-zu-datenserviceinformationen>.

Die Frage der Registrierung im ambulanten Bereich ist bei Bedarf mit der entsprechenden Datenannahmestelle abzustimmen.

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

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.

Mit der Spezifikation 2020 wurde zusätzlich ein 36-stelliger pseudozufälliger Globally Unique Identifier (GUID) eingeführt, der die Datensätze bundesweit eindeutig identifiziert ohne Rückschlüsse auf den Leistungserbringer oder Patienten zuzulassen. Dadurch ist es nicht mehr nötig, dass die Datenannahmestellen aus den Vorgangsnummern bundesweit eindeutige Vorgangsnummern machen.



Achtung

Die Dokumentationssoftware muss sicherstellen, dass die `VorgangsnrGuid` bundesweit eindeutig einem Behandlungsfall für ein Exportmodul darstellt. Die eindeutige GUID darf weder verschiedenen Exportmodulen noch verschiedenen Leistungserbringern zugeordnet sein.

Annahme oder Ablehnung unterschiedlicher Versionen eines Datensatzes

In den Datenannahmestellen können Vorgänge/Fälle anhand der `VorgangsnrGuid` oder der Kombination aus Registrierungsnummer und Vorgangsnummer identifiziert werden. Eine eindeutige und dauerhafte Identifizierung eines Vorganges im gesamten Datenfluss ist aber nur mit der `VorgangsnrGuid` möglich.

Beim Leistungserbringer und Dateneingang der Datenannahmestellen muss gelten, dass Datensätze mit identischen Kombinationen aus Registriernummer und Vorgangsnummer immer auf dieselbe `VorgangsnrGuid` verweisen. Diese Beziehung soll eineindeutig sein, das heißt, Datensätze mit identischen `VorgangsnrGuids` müssen umgekehrt immer auf dieselben Kombinationen aus Registriernummer und Vorgangsnummer verweisen. Ab Datenausgang der Datenannahmestellen

gilt dagegen, dass eine VorgangsnrGuid nur noch mit der gleichen Vorgangsnummer assoziiert sein muss und nicht mehr mit der Kombination aus Vorgangsnummer und Registriernummer, denn in der Datenannahmestelle wird die Registriernummer der Leistungserbringer mit der Registriernummer der Datenannahmestelle ausgetauscht.

Der für einen bestimmten Vorgang gespeicherte Datensatz kann nur durch eine neuere Version (mit höherer Versionsnummer) überschrieben werden.¹⁸ Unterschiedliche Versionen eines Datensatzes müssen demselben Primärmodul¹⁹ zugeordnet sein. Ein Datensatz mit einer Vorgangsnr-Guids, die bereits unter einem anderen Modul eingeschickt wurde, wird abgelehnt.

Ein Minimaldatensatz darf nur für Primärmodule angelegt 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

Tabelle 16: Benennungselemente der Exportdateien

Element	Bedeutung
GUID	Die verwendete GUID ist die im Dokument verwendete ID des Datenpakets (Abschnitt A „Eindeutige Kennzeichnung der XML-Exportdateien“ auf S. 77).
Inhaltskennung	Q → QS-Daten S → Sollstatistik R → Routinedaten
Protokolltyp	T → Transaktionsprotokoll bzw. Empfangsbestätigung D → Datenflussprotokoll
Rolle Absender	LE → Leistungserbringer DAS → Datenannahmestelle VST → Vertrauensstelle VPB → Versendestelle Patientenbefragung BAS → Bundesauswertungsstelle

Beispiele:

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

(QS-Daten eines Leistungserbringers)

¹⁸ Gegebenenfalls ist der geänderte Datensatz mit einer neuen Versionsnummer zu übermitteln.

¹⁹ Jeder Datensatz ist einem Primärmodul zugeordnet. Auch dem Minimaldatensatz (MDS) ist ein Primärmodul zugeordnet (Bogenfeld ZUQSMODUL).

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)

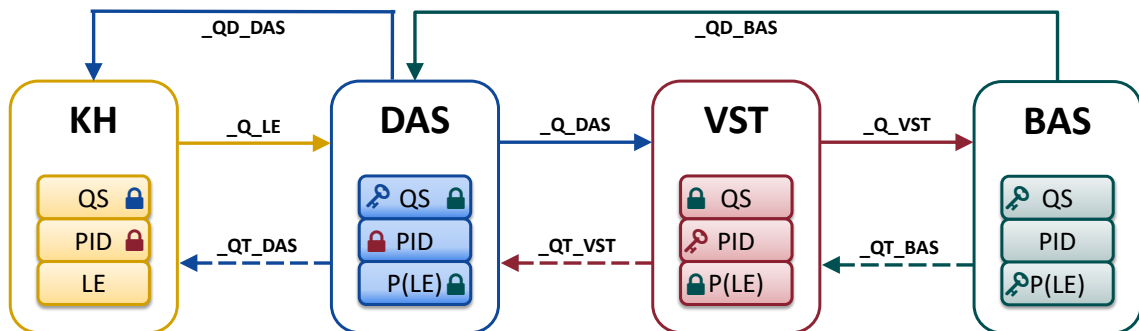


Abbildung 5: Übersicht über die einzusetzenden Suffixe

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 Dokumentenabschnitte, 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.

Ausgangsvalidierung 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 Ausgangsvalidierung:

- 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.²⁰ 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

T-NI1234A-2024_09_19_160945.zip.aes

T-BW1234a-2024_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.

²⁰ <http://www.w3.org/XML/Schema>.

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-2024_09_23_094051.zip.aes  
auf die Transaktionsdatei: T-BW1234a-2024_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“ auf S. 77) 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.

5.2.2 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, KIM 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 Spezifikationsjahres (Release V01) bereitzustellen. Im Rahmen aller weiteren Releases für ein Spezifikationsjahr 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:

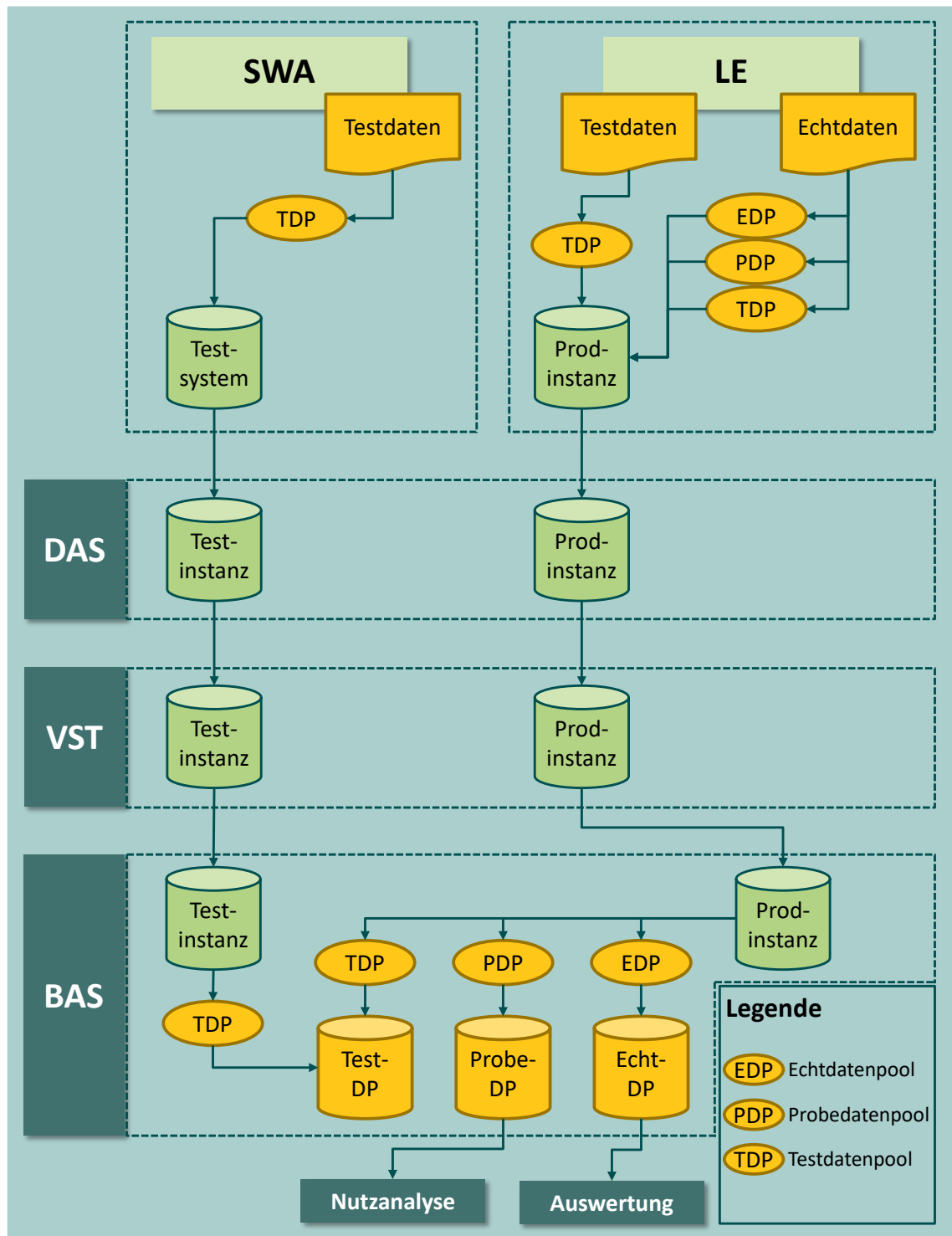


Abbildung 6: Datenflüsse im Test-, Probe- und Regelbetrieb am Bsp. der Follow-up-Verfahren

Auf der Testinstanz können Prüfungen deaktiviert sein.

Testbetrieb mit Testsystemen und Testungen mit Produktivinstanzen

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

**Achtung**

Beim Testbetrieb mit Testsystemen dürfen keine Echtdaten 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 „“ auf S. **Fehler! Textmarke nicht definiert.**). 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 Echtdaten durchgeführt werden. Dazu muss zwingend das Datenziel „Testdatenpool“ angegeben werden.

Probetrieb

Probetriebe erfordern eine eigene Spezifikation.

Regelbetrieb

Der Regelbetrieb erfordert eine eigene Spezifikation.

5.3 Rückprotokollierung

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

5.3.1 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.

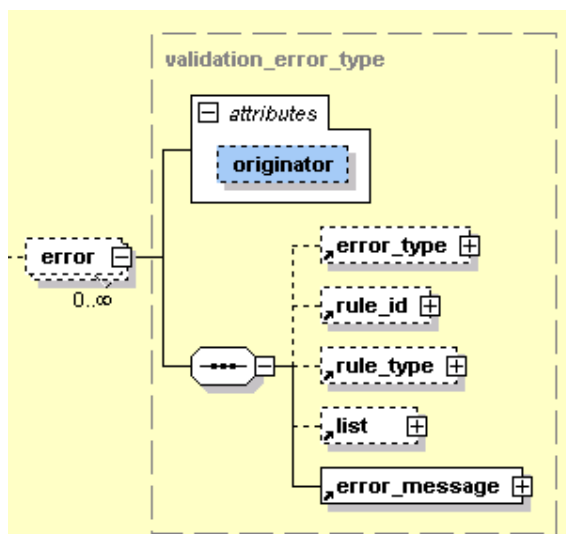


Abbildung 7: Attribut „originator“ im Prüfungs- und Fehlerprotokoll

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 verfahrenssupport@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.

5.3.2 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" container version="2.0" content_version="1.0" xsi:noNamespaceSchemaLocation="../../interface_LE_DAS/response_receipt.xsd">
  <!-- xsi:schemaLocation ist optional und hier nur als Hinweis (IDE/Editor-Hilfe) zu verstehen, welches Schema zu diesem Beispiel gehört. -->
  <header>
    <document>
      <id V="{f83f11a8-7d45-4f81-961c-dbef7e14fa1e}" />
    </document>
    <provider address="Virchowstraße 1, 10123 Musterstadt" email="email@domain.tld" fax="0123456789" function="Datenannahmestelle" name="Datenannahmestelle Beispielname" phone="0123456789" registration="bu365478" />
  </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 Auswertungsstelle wird folgen. Erst das Datenflussprotokoll informiert über die Prüfergebnisse. Valide Datensätze gelten als geliefert, fehlerhafte Datensätze müssen korrigiert und erneut gesendet werden.</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 wird folgen.

Erst das Datenflussprotokoll informiert über die Prüfergebnisse. Valide Datensätze gelten als geliefert, fehlerhafte Datensätze müssen korrigiert und erneut gesendet werden.

Die Benennung der unverschlüsselten Datei leitet sich aus dem Benennungsschema ab (Abschnitt A „Benennung der Exportdateien“ in 5.2.1).

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“ auf S. 96 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 17 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 Teildatensatzes (Beispiele: `B/BEGRLTP`). Der Bezug zum Modul kann entfallen, da dieses über die Vorgangsnummer identifiziert werden kann.

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.

^^Tabelle 17: Ausfüllen der Elemente eines Validation-Items in Abhängigkeit von den Fehlerarten

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

Feld (csv)	Fehlerart	Regelnr	Regeltyp	Liste	Meldung
Element (xml)	<error_type>	<rule_id>	<rule_type>	<list>	<error_message>
	WERT	Ja	-	<Bogenfeldliste> ²¹	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>

```

Um die Fehleranalyse zu vereinfachen, wurden die potenziellen Fehler in Kategorien (XML-Element <error_type>) unterteilt, die bestimmte Prüfprozesse (XML-Element <validation_item>) durchlaufen. 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 das Beispiel bei einem bestimmten Fehlerfall.

Tabelle 18: Beispiel für eine Fehlermeldungen

Fehlerart	Standardisierte Meldung	Beschreibung
Standardisierung der Meldungen bei Bestätigungstatus mit Fehlerart DOP-PELT.	Es wurde bereits ein Datensatz mit derselben Registrier- und Vorgangsnummer und derselben oder einer höheren Versionsnummer übermittelt.	Bei feldbezogenen Fehlern sind die standardisierten Fehlermeldungen der Plausibilitätsregeln zu verwenden.

²¹ In der Regel wird hier nur ein Bogenfeld aufgeführt. Ausnahme ist, wenn Kombinationsfelder geprüft werden: ENTLDIAG | ENTDIAGVERS u.a.

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.

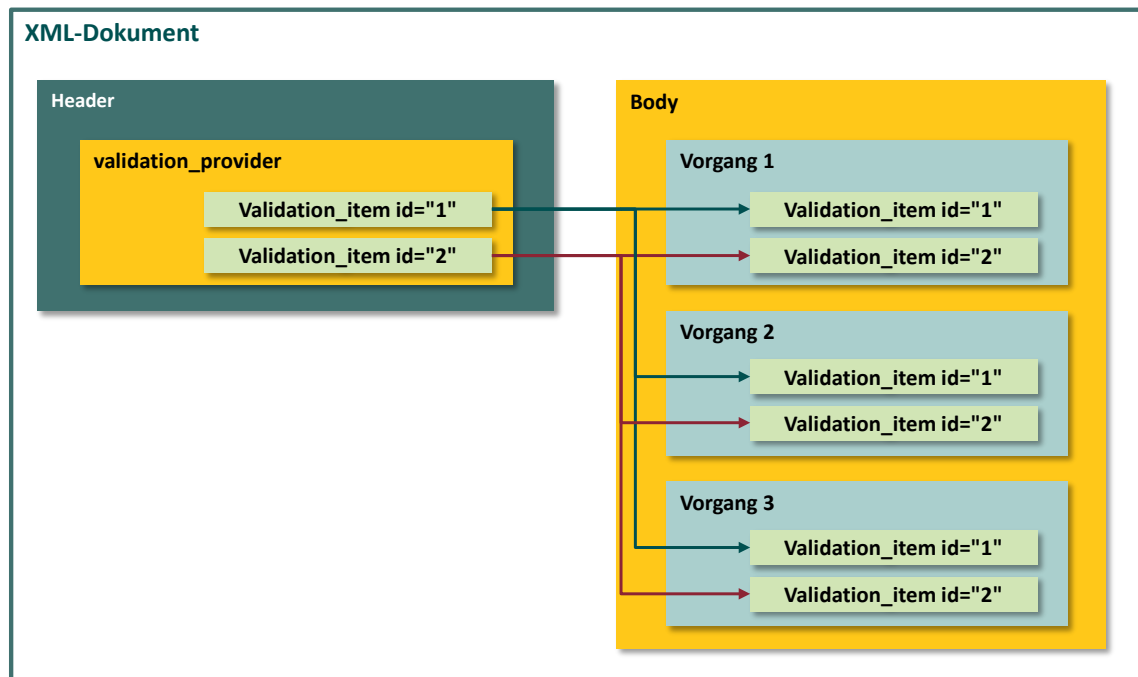


Abbildung 9: Beziehungen zwischen `<validation_item>` im header und `<validation_item>` im body über die id

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 2.7.3).

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.

5.4 Meldung eines Leistungserbringermappings

Nur im Fall einer Rechtsnachfolge muss der Leistungserbringer das Leistungserbringermapping (LE-Mapping) ausdrücklich an die für ihn zuständige DAS kommunizieren. Die DAS leitet dann das LE-Mapping weiter an die BAS. Für die Meldung des LE-Mapping an das IQTIG wird das Pseudonymisierungsprogramm (PSP) verwendet.

Wenn keine Rechtsnachfolge vorliegt, erfolgt kein LE-Mapping und der Wechsel muss nicht an die BAS über die DAS kommuniziert werden. Damit liegen zwei unterschiedliche Betriebsstätten vor, die rechtlich getrennt behandelt werden und damit erhält jede Betriebsstätte eigene Rückmeldeberichte.

B Komponenten

In diesem Kapitel werden die einzelnen Komponenten der Spezifikation beschrieben. Ein Spezifikationspaket bildet die Gesamtheit seiner Spezifikationskomponenten ab. Hierbei handelt es sich nur um die Beschreibung der Komponenten der Basisspezifikation, die relevant sind für das Verfahren Ambulante Psychotherapie.

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
 - **TechDok_Patientenbefragung** – Technische Dokumentation für die Patientenbefragung (PPCI)
 - **TechDok_Benennungsschema** – Technische Dokumentation für das Benennungsschema
 - **TechDok_Ambulante Psychotherapie** – Technische Dokumentation spezifisch für das Verfahren Ambulante Psychotherapie
- **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 Datenbankinhalte²² 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.
- **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.
- **UebersichtAenderungen** – ist eine PDF-Datei mit der Übersicht über die Änderungen in der Spezifikation zu allen Vorversionen des Spezifikationsjahres 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).
- **Komponentenuebersicht** – 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 ist zur Zeit das Datenprüfprogramm 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.

- **Datenpruefprogramm** – ist auf Komponentenebene eine ZIP-Datei, welche die Bestandteile des Datenprüfprogramms enthält.

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

²² 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.

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 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 zu Datenserviceinformationen

Seit der Spezifikation 2019 werden relevante Datenserviceinformationen (Abschnitt 0) 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.

Verschlüsselungsprogramme

Seit der Spezifikation 2020 werden die Verschlüsselungsprogramme nicht mehr als Spezifikationskomponente veröffentlicht. Sie sind daher keine dem Spezifikationspaket zugehörige Komponente, verfügen jedoch über dieselbe Verbindlichkeit. Da sie ein eigenständiges Paket in Form einer ZIP-Datei darstellen, können sie außerhalb des Releasezyklus angepasst werden.

In der ZIP-Datei enthalten sind der X- und der TPacker sowie die zum Veröffentlichungszeitpunkt aktuellen öffentlichen Schlüssel der Datenservices im Datenfluss.

Pseudonymisierungsprogramm

Seit der Spezifikation 2021 wird das Pseudonymisierungsprogramm nicht mehr als Spezifikationskomponente veröffentlicht. Sie ist daher keine dem Spezifikationspaket zugehörige Komponente, verfügt jedoch über dieselbe Verbindlichkeit. Sie kann außerhalb des Releasezyklus zur Verfügung gestellt werden.

- **PSP** – ist eine ZIP-Datei, die neben dem Pseudonymisierungsprogramm ein Informationsmerkblatt beinhaltet.

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 Tabellename 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 Attribut²³, 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 `KURZER TEXT (1)` (entspricht `VARCHAR(1)`). Diese Tabellen sind als einfache „Nachschlagtabellen“ zu interpretieren. Im Fall der Tabelle `MussKann` soll im entsprechenden Fremdschlüsselfeld der verknüpften

²³ Oder sie enthalten eine identifizierende Attributkombination, die einen eindeutigen Schlüssel definiert.

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 Primärschlüssel²⁴, jedoch einen eindeutigen Schlüssel, der über die Fremdschlüsselfelder definiert ist.

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 `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`.

²⁴ Hier: Primärschlüssel im Sinne der Access-Definition eines Primärschlüssels. Streng genommen wird über die beiden Fremdschlüssel ein neuer Primärschlüssel definiert.

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 zur Verfügung gestellt. Der Name der QS-Filter-Spezifikation richtet sich nach folgendem Schema:

`<Spezifikationsjahr>_BASIS_QSF_V<Versionsnummer>.mdb`

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

Beispiel:

Im Spezifikationsjahr 2027 ist die QS-Filter-Spezifikation 2027_BASIS_QSF_V01.mdb²⁵ gültig. Die Kennung 2027 gilt für das Spezifikationsjahr 2027.

Weiterführende Erläuterungen zum Benennungsschema für Spezifikationskomponenten sind der Technischen Dokumentation für das Benennungsschema zu entnehmen.

Die Tabellenstruktur der Spezifikationsdatenbank wird in Abschnitt B 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 QS-Verfahren 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.

²⁵ Die Versionsnummer der gültigen Spezifikation (z.B. V01, V02 usw.) ist dem zuletzt veröffentlichten Update zu entnehmen.

- OPS-Listen
Die OPS-Listen geben die Einschluss- und Ausschlussprozeduren für jedes Modul an.
- Auslösebedingungen
Diese Abfrage liefert einen Überblick über die in der Spezifikation enthaltenen Auslösebedingungen der QS-Filter-Leistungsbereiche.
- 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_DeQS_KV
Diese Abfrage liefert einen Überblick über den Datensatz SOLL_DeQS_KV
- SOLL_DeQS_LKG
Diese Abfrage liefert einen Überblick über den Datensatz SOLL_DeQS_LKG
- SOLL_DeQS_SV
Diese Abfrage liefert einen Überblick über den Datensatz SOLL_DeQS_SV

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.²⁶ 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.

Beispiel:

Der Modulauslöser `APSY` löst den QS-Datensatz `APSY` aus. Der Datensatz wird als Exportmodul `APSY_KV` oder `APSY_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 2.2.1 beschrieben.

²⁶ Landesweit verpflichtende Module haben keinen Datensatz in der Spezifikation für QS-Dokumentationssoftware.

Tabelle 19: Struktur der Tabelle Modul

Feldname	Datentyp	Bemerkung
idModul	INTEGER	Primärschlüssel
name	KURZER TEXT	Technischer Name des Moduls (Identifizierendes Attribut)
bezeichnung	KURZER TEXT	Beschreibender Text für den QS-Datensatz
fkSchluesselWert	INTEGER	Verweis auf denjenigen Kode des Schlüssels Modul (Tabelle SchluesselWert), welcher dem betreffenden Datensatz zugeordnet ist. Der Kode wird in der Datei SOLL-MODUL_DeQS_<Spezifikationsjahr>.txt der elektronischen Sollstatistik verwendet.
fkModulParent	INTEGER	Verweis auf das Eltern-Modul eines Zählleistungsbereichs
zaehlLb	BOOLEAN	Wenn WAHR, existiert kein entsprechender QS-Datensatz, es handelt sich um einen „Zählleistungsbereich“ zur separaten Darstellung in der Sollstatistik.
ausloeseModul	BOOLEAN	Wenn WAHR, existiert ein entsprechender QS-Datensatz, der ausgelöst werden kann. Dieses Modul kann gleich dem Exportmodul sein, z. B. HCH.
exportModul	BOOLEAN	Wenn WAHR, handelt es sich um ein Exportmodul, z. B. PCI_LKG. Das Exportmodul kann ungleich dem im Modulauslöser referenzierten Modul sein, z. B. PCI.

1.2.2 Struktur der Datensatzdefinitionen

Die Module werden über den definierten Modulauslöser als dokumentationspflichtig erkannt, indem die hinterlegte Bedingung mit den im System (z.B. PVS) gespeicherten Daten geprüft wird. Ist die Bedingung erfüllt, wird das Modul ausgelöst. Da die in der Bedingung enthaltenen Felder im System (z.B. PVS) 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)

r jeden dieser Datensätze wird in der Tabelle D_s ein Eintrag angelegt.

Tabelle 20: Struktur der Tabelle D_s

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

Teildatensätze

Die Definition von Teildatensätzen befindet sich in der Tabelle $T_d s$ der Datenbank (Tabelle 21). Jeder Teildatensatz ist eindeutig durch seinen Namen (z. B. FALL, PROZ oder DIAG) charakterisiert.

Tabelle 21: Struktur der Tabelle $T_d s$

Feldname	Datentyp	Bemerkung
idTds	INTEGER	Primärschlüssel
name	KURZER TEXT	Technischer Name des Teildatensatzes (Identifizierendes Attribut)
bezeichnung	KURZER TEXT	Beschreibender Text
fkTds	INTEGER	Optionaler Fremdschlüssel zu einem Mutterteildatensatz
fkDs	INTEGER	Bezug des Teildatensatzes zum übergeordneten Datensatz in der Tabelle D_s , z. B. QS-Filter-Datensatz oder QS-Filter-Datensatz für Sollstatistik
fkRelation-Typ	KURZER 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 $T_d s$ 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: Teildatensatz²⁷ $fkTds = NULL$ in der Tabelle $T_d s$). Über

²⁷ Es darf nur eine Definition eines Basisteildatensatzes existieren.

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)

Beispiele:

Der QS-Filter-Datensatz `ANWBEZOGEN` besitzt den Basisteildatensatz `FALL`.

Felder der Teildatensätze

Die Felder eines Teildatensatzes sind in der Tabelle `TdsFeld` 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 22: Struktur der Tabelle `BasisTyp`

Feldname	Datentyp	Bemerkung
<code>idBasisTyp</code>	<code>INTEGER</code>	Primärschlüssel
<code>name</code>	<code>KURZER TEXT</code>	Technischer Name (muss eindeutig sein)
<code>bezeichnung</code>	<code>KURZER TEXT</code>	Beschreibender Text
<code>formatAnweisung</code>	<code>KURZER TEXT</code>	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 `SchluesseelWert` definiert. Externe Schlüsselkataloge (z. B. OPS, EBM oder ICD10) sind von den entsprechenden Anbietern zu beziehen.

Tabelle 23: Struktur der Tabelle `Schluesseel`

Feldname	Datentyp	Bemerkung
<code>idSchluesseel</code>	INTEGER	Primärschlüssel
<code>name</code>	KURZER TEXT	Technischer Name (muss eindeutig sein)
<code>bezeichnung</code>	KURZER TEXT	Beschreibender Text
<code>extern</code>	BOOLEAN	Zeigt an, ob der Schlüssel in der Tabelle <code>Schluesseel</code> oder in einer externen Tabelle gespeichert ist.
<code>externVerweis</code>	KURZER 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.bfarm.de>). Ein Verweis auf eine Bezugsquelle kann unabhängig vom Attribut `extern` angegeben werden (siehe Schlüssel `EntlGrund`).



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 24: Struktur der Tabelle `SchluesselWert`

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



Hinweis

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

1.2.3 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.

Der aktuelle Stand der in der Syntax verwendeten Funktionen ist in der Tabelle `SyntaxFunktion` der Spezifikation zu finden.

Weiterführende Informationen zur Implementierung von Funktionen sind Abschnitt B 2.3.3 zu entnehmen.

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 2.4.2).



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 sind die Auslösung nur mithilfe von gesicherten Diagnosen zulässig.

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:

`APSY_EBM` Einschlussgebührenordnungspositionen ambulante Psychotherapie

String-Vergleich bei EBM-Kodes

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

In der Spezifikationsdatenbank für QS-Filtersoftware sind die EBM-Kodes in Tabelle `EBMWert`

fünfstellig – ohne ergänzende Zeichen – definiert. Für die Prüfung, ob zwei Codes identisch sind, genügt kein einfacher Stringvergleich. Stattdessen wird ein Stringvergleich der Normcodes²⁸ durchgeführt, um die Übereinstimmung zwischen dem dokumentierten Kode und dem einer EBM-Liste zu ermitteln.

1.2.7 Versionsverwaltung

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

Ab der Spezifikation 2022 bildet die Tabelle `Version` ausschließlich die Einträge zum aktuellen Spezifikationsjahr der Spezifikation ab. Einträge zu früheren Spezifikationsjahren werden gelöscht.

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 Spezifikationsjahr (z. B. Aufnahme zwischen dem 01.01.2027 und dem 31.12.2027).

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 Versionen²⁹: Welche Vorgängerversion vorher gültig war, kann über das Attribut `fkVersion` ermittelt werden.

Tabelle 25: Struktur der Tabelle `Version`

Feldname	Datentyp	Bemerkung
<code>idVersion</code>	INTEGER	Primärschlüssel
<code>name</code>	KURZER TEXT	Technischer Name der Version (Identifizierendes Attribut)
<code>bezeichnung</code>	KURZER 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

²⁸ Jeder EBM-Kode lässt sich entweder als Kode mit ergänzenden Zeichen (Normcode + ergänzende Zeichen) oder als Kode ohne ergänzende Zeichen (Normcode) darstellen.

²⁹ Die Inhalte der Vorversionen sind nicht Teil der aktuellen Spezifikationsdatenbank.

Feldname	Datentyp	Bemerkung
fkVersStatus	CHAR	Bezug zum Status einer Version (Tabelle VersStatus): 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 Systempflege³⁰ 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.

Zuordnung der QS-Filter-Version zu Behandlungsfällen

Die QS-Filter-Software eines Spezifikationsjahres wird für Behandlungsfälle verwendet, deren Aufnahmedatum 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 10 stellt dar, für welche stationären Behandlungsfälle welche Version der QS-Filter-Software verwendet wird. Exemplarische Behandlungsfälle sind durch Querbalken visualisiert, welche den Behandlungsabschnitt vom Aufnahmedatum bis zum Entlassungsdatum des Krankenhausaufenthalts abdecken. In Abhängigkeit vom Aufnahmedatum 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.

³⁰ Die Aktualisierung der ICD- und OPS-Kodes erfolgt auch für nicht verpflichtende Module weiterhin im Rahmen des BfArM-Umstiegs.

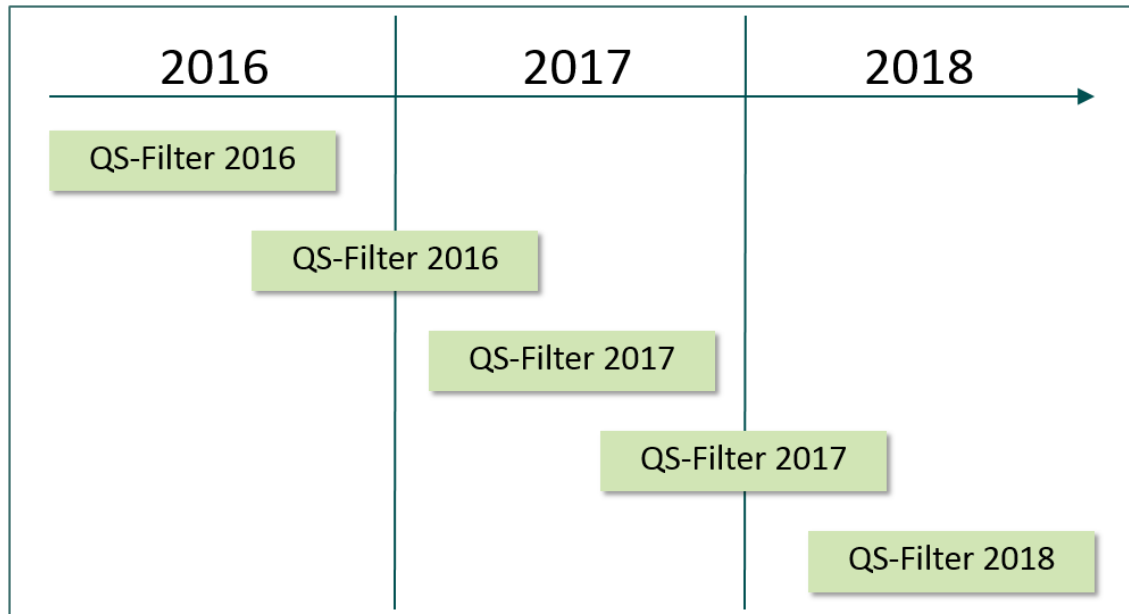


Abbildung 10: Zuordnung der Version des QS-Filters zu den Behandlungsfällen: Kriterium ist das Aufnahmedatum

1.2.8 Meta-Tabellen

In den Tabellen `TabellenStruktur` und `TabellenFeldStruktur` werden die Tabellen und ihre Attribute aufgelistet. Der Inhalt wird automatisch generiert (Abschnitt B 2.6.2).

1.2.9 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 2.6.2).

Beispiel (Spezifikation 2027):

Die Datenbank `2027_BASIS_QSF_V01` stellt die Änderungen im Vergleich zur letzten gültigen Spezifikationsdatenbank des Vorjahres `2026_BASIS_QSF_V06` dar.

Die Datenbank `2027_BASIS_QSF_V02` stellt zusätzlich die Änderungen im Vergleich zur Vorversion `2027_BASIS_QSF_V01` dar.

1.3 Der QS-Filter-Datensatz

Der QS-Filter-Datensatz des QS-Verfahrens Ambulante Psychotherapie umfasst den QS-Filter-Eingangsdatensatz und die Anwenderbezogene Fallidentifikation. Verbindlich für ein Spezifikationsjahr ist die Datenfeldbeschreibung, welche in der gültigen Spezifikationsdatenbank über die Abfrage `Datensatz QS-Filter` definiert ist.

Hier werden die für das QS-Verfahren ambulante Psychotherapie relevanten Datensätze dargestellt.

Tabelle 26: 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
9	ANWBEZOGEN	Anwenderbezogene Fallidentifikation

**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 DeQS-RL für durch ein Krankenhaus erbrachte Fälle finden sich im Aufnahmedatensatz bzw. in der Entlassungsanzeige des § 301-Datenformats³¹. Daher ist der QS-Filter-Eingangsdatensatz nach § 301 Abs. 3 SGB V (DATENSATZ_301) so beschaffen, dass fast alle seine 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)

³¹ Datenübermittlung nach §301 Abs. 3 SGB V, ergänzt um die Festlegungen der Vereinbarung nach §120 Abs. 3 SGB V

Die hier definierten Teildatensätze finden sich im § 301-Datensatz bzw. dem DRG-Datenformat nach § 21 KHEntgG ³² wieder. Der QS-Filter-Eingangsdatensatz basiert zum größten Teil auf einer Teilmenge der Inhalte dieser Datensätze.

Tabelle 27: Ausgewählte 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)
GEBDATUM	Geburtsdatum	M	10	DATUM
PATALTER	Alter in Jahren am Aufnahme- tag	M	3	GANZEZAHL
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	9	TEXT

³² 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.

KASSEIKNR	Institutionskennzeichen der Krankenkasse der Versichertenkarte	K	9	TEXT
VERSI-CHERTENIDNEU	eGK-Versichertennummer	K	10	TEXT
PERSONENKREIS	besonderer Personenkreis	K	2	SCHLUESSEL (Personenkreis)
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)
Entgeltangaben (wiederholbar) (ENTGELT)				
FALLNUMMER	Fallnummer	M	15	TEXT
ENTGELTARTS-TAT	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
FABENTLDDATUM	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 sind³³. Die nachfolgend aufgelisteten Codes sind für die Zuordnung der Versorgungsarten relevant:

- Kode 70³⁴: DRG-Fallpauschale nach § 7 Nr. 1 KHEntgG (§ 1 Abs. 1 Satz 1 KVPV/FPV)
- Kode 61³⁵: Entgelt für integrierte Versorgung nach § 140c SGB V
- Kode 65³⁶: Zusatzentgelt für DMP
- Sonstiger Fall³⁷: Fall, der weder ein DRG-Fall noch ein IV-Fall noch ein DMP-Fall ist (z. B. ein ambulanter Behandlungsfall oder ein Fall, der auf Basis der BPfIV nach Tagessätzen abgerechnet wird).

Die Details der Zuordnung sind in Abschnitt B 0 beschrieben.

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 (z.B. Praxen oder MVZ)³⁸ 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 können³⁹. 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)

Tabelle 28: Ausgewählte Felder des QS-Filter-Eingangsdatensatzes nach § 295

Feld	Beschreibung	M/K	Zeichenlängen	BasisTyp
Behandlungsfall (<code>FALL</code>)				

³³ Aktuelle Version abrufbar im Download-Bereich der DKG-Homepage (<http://www.dkgev.de>)

³⁴ Versorgungsart DRGFALL (Datenfeld des QS-Filter-Ausgangsdatensatz)

³⁵ Versorgungsart IVFALL (Datenfeld des QS-Filter-Ausgangsdatensatz)

³⁶ Versorgungsart DMPFALL (Datenfeld des QS-Filter-Ausgangsdatensatz)

³⁷ Versorgungsart SONSTFALL (Datenfeld des QS-Filter-Ausgangsdatensatz)

³⁸ Ambulante Leistungen nach § 295 SGB V, § 116 SGB V, stationäre Fälle nach § 121 Abs. 3 SGB.

³⁹ 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).

Feld	Beschreibung	M/K	Zeichenlänge	BasisTyp
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	TEXT
KASSEIKNR	Kostenträgernummer	M	9	TEXT
VERSICHERTENIDNEU	eGK-Versichertennummer	K	10	TEXT
PERSONENKREIS	besonderer Personenkreis	K	2	SCHLUESSEL (Personenkreis)
Diagnoseangaben (wiederholbar) (DIAG)				
BEHANDLFALLID	Behandlungsfall-ID	M	15	TEXT
ICD	ICD-Kode	M	9	SCHLUESSEL (ICD)
DIAGSICHERHEIT	Diagnosesicherheit	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 vorliegt⁴⁰ 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.

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 Kodes bzw. Felder identifiziert und ergänzend im QS-Filter-Eingangsdatensatz berücksichtigt und/oder auf anderen Wegen für die Auslösung und/oder Dokumentation zur Verfügung gestellt werden.

⁴⁰ Da die Abrechnungsgrundlagen für selektivvertraglich erbrachte ambulante Leistungen nicht bundesweit eindeutig definiert sind, wird der QS-Filter-Datensatz für diese Fälle möglichst breit spezifiziert.

1.3.2 Der Datensatz „Anwenderbezogene Fallidentifikation“

Der Datensatz ANWBEZOGEN („Anwenderbezogene Fallidentifikation“) stellt eine spezifische Neuerung für das QS-Verfahren Ambulante Psychotherapie dar.

Er definiert Felder, die zur Auslösung der Dokumentationspflicht von der Anwenderin oder vom Anwender angegeben werden müssen. Diese Felder werden in Teilbedingungen des Modulauslösers APSY und PAPSY überprüft. Sie ersetzen eine automatisierte Überprüfung ausschließlich über Felder der anderen Eingangsdatenätze, da diese sich bei der Einführung des Verfahrens als nicht praktikabel erwiesen.

Tabelle 29: Felder des QS-Filterdatensatzes "Anwenderbezogene Fallidentifikation"

Feld	Beschreibung	M/K	Zeichenlänge	BasisTyp
GRUPPENTH	Gruppentherapie	K	1	NUMSCHLUESSEL (JN)
KJPSYT	Kinder- und Jugendlichenpsychotherapie	K	1	NUMSCHLUESSEL (JN)
BEGRLTP	Beginn der Richtlinien- therapie nach dem 31.08.2024	K	1	NUMSCHLUESSEL (JN)
ALTERBEG	Alter zu Beginn mindes- tens 18 Jahre	K	1	NUMSCHLUESSEL (JN)

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 leistungsbezogenen 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 30) 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 APSY)

```
VERSICHERTENIDNEU <> LEER UND format(VERSICHERTENIDNEU; '[A-Z][0-9]{9}') = WAHR UND LENGTH(KASSEIKNR) = 9 UND LEFT(KASSEIKNR;2) = '10'
UND PERSONENKREIS = '00' UND EBM EINSIN APSY_EBM UND TeilStr(LANR;8;2) IN ('51';'58';'60';'61';'68') UND GRUPPENTH = 0 UND
KJPSYT = 0 UND BEGRLTP = 1 UND ALTERBEG = 1
```

setzt sich aus folgenden Teilbedingungen zusammen:

- `VERSICHERTENIDNEU <> LEER UND format(VERSICHERTENIDNEU; '[A-Z][0-9]{9}')`
- `LENGTH(KASSEIKNR) = 9 UND LEFT(KASSEIKNR;2) = '10'`
- `PERSONENKREIS = '00'`
- `EBM EINSIN APSY_EBM`
- `TeilStr(LANR;8;2) IN ('51';'58';'60';'61';'68')`

- GRUPPENTH = 0
- KJPSYT = 0
- BEGRLTP = 1
- ALTERBEG = 1

APSY_EBM stellt die Einschluss- bzw. Ausschlussliste dar, die in den Tabellen ICDListe bzw. OPSListe spezifiziert wurde und deren Schlüsselwerte in den Tabellen ICDWert bzw. OPSWert zu finden sind.

**Achtung****Einschlusskriterien mit KASSEIKNR, VERSICHERTENIDNEU und PERSONENKREIS**

Verfahrensbezogene Einschlusskriterien können die Felder KASSEIKNR, VERSICHERTENIDNEU und PERSONENKREIS enthalten, mithilfe deren 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.

Definition in der Datenbank

Tabelle 30 gibt einen Überblick über die Tabelle `ModulAusloeser` der Spezifikationsdatenbank.

Tabelle 30: Struktur der Tabelle `ModulAusloeser`

Feldname	Datentyp	Bemerkung
<code>idModulAusloeser</code>	INTEGER	Primärschlüssel
<code>name</code>	KURZER TEXT	Technischer Name des QS-Filter-Leistungsbereichs
<code>bezeichnung</code>	KURZER TEXT	Bezeichnungstext der Auslösebedingung
<code>bedingung</code>	LANGER TEXT	Auslösebedingung für den QS-Filter-Leistungsbereich
<code>textDefinition</code>	LANGER TEXT	Medizinisch-inhaltliche Definition bzw. Erläuterung der Auslösebedingung
<code>fkModul</code>	INTEGER	Bezug zum Modul
<code>verpflichtend</code>	BOOLEAN	Wenn WAHR, ist das betreffende Modul bundesweit verpflichtend
<code>fkAdminKriterium</code>	INTEGER	Bezug zu den administrativen Abgrenzungskriterien
<code>fkDs</code>	INTEGER	Bezug zum entsprechenden Sollstatistikdatensatz
<code>internBild</code>	KURZER TEXT	Bild zum Modul
<code>ambulant</code>	BOOLEAN	Ist die Fallart ambulant?
<code>stationaer</code>	BOOLEAN	Ist die Fallart stationär?
<code>krankenhausabrechnung</code>	BOOLEAN	Wird der Fall stationär abgerechnet?
<code>selektiv</code>	BOOLEAN	Wird der Fall ambulant selektivvertraglich abgerechnet?

Feldname	Datentyp	Bemerkung
kollektiv	BOOLEAN	Wird der Fall ambulant kollektivvertraglich abgerechnet?

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'; '10'; '11') UND AUFNGRUND <> LEER

Diese Teilbedingung ist Teil der administrativen Kriterien (siehe Attribut `bedingung` in Tabelle 31). Der zeitliche Rahmen wird bei stationären Fällen über das Aufnahme- und das Entlassungsdatum definiert.

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 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.

Definition in der Datenbank

Tabelle 31 gibt einen Überblick über die Tabelle `AdminKriterium`, welche die administrativen Abgrenzungskriterien definiert.

Tabelle 31: Struktur der Tabelle AdminKriterium

Feldname	Datentyp	Bemerkung
idAdminKriterium	INTEGER	Primärschlüssel
name	KURZER TEXT	Technischer Name des administrativen Kriteriums
bezeichnung	KURZER TEXT	Bezeichnung des administrativen Kriteriums
bedingung	LANGER TEXT	Auslösebedingung für das QS-Filter-Modul
info	LANGER TEXT	Bezeichnungstext des administrativen Kriteriums
sollBedingung	LANGER TEXT	Auslösebedingung für die Sollstatistik des aktuellen Spezifikationsjahres (enthält nur dann Informationen, wenn die Auslösebedingung der Sollstatistik von der Bedingung des QS-Filters zur Auslösung der Dokumentationspflicht abweicht)
sollBedingungUeberlie- ger	LANGER TEXT	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 33).

Typen

Jede Variable hat einen Basistyp. Tabelle 32 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 32: Basistypen der Variablen

Basistyp	Bezeichnung	Beispiele (Literale)
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'
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.

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 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.

Die Operatoren der Auslösebedingungen

Tabelle 33 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 33: Präzedenz und Assoziativität der Operatoren⁴¹

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)⁴².

⁴¹ 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.

⁴² Dieser Operator ist redundant, da er auch durch Negation des **EINSIN**-Operators abgedeckt ist.

Beispiel:

Folgende Regel prüft, ob ein Element der Listenvariable `EBM` einen bestimmten Kode besitzt:

```
EBM EINSIN APSY_EBM
```

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')
```

1.4.5 Stufen der Dokumentationsverpflichtung

Die Spezifikation für QS-Filter enthält:

- bundesweit **verpflichtende** (dokumentationspflichtige) QS-Verfahren,
- bundesweit **freiwillige** QS-Verfahren, z. B. freiwillige Verfahren oder landesweit verpflichtende QS-Verfahren

Bundesweit dokumentationspflichtige QS-Verfahren sind erkennbar am Attribut `verpflichtend` in der Tabelle `ModulAusloeser`.

Bundesweit freiwillig zu dokumentierende QS-Verfahren 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 QS-Verfahren wird in jeder Installation von QS-Filter-Software konfiguriert.

In Tabelle 34 sind die Stufen der Dokumentationsverpflichtung (Schlüssel `DokVerpflicht` der Datenbank) aufgelistet.

Tabelle 34: 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	Zähleleistungsbereich (bundesweit erhoben, Dokumentation erfolgt im Elternmodul)	ja

Die Stufe der Dokumentationsverpflichtung wird vor Beginn des Spezifikationsjahres in der QS-Filter-Software eines Krankenhauses für alle QS-Verfahren konfiguriert.

Die Konfiguration der Stufen darf im Laufe des Spezifikationsjahres nicht mehr verändert werden, da sonst die Vollständigkeit der Dokumentationen nicht mehr gewährleistet ist.

Für jedes ausgelöste QS-Verfahren 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 35: Fehlercodes des QS-Filters

Beschreibung des Fehlers	Fehler-kode	Formale Fehlerbedingung	Fehlermeldung
Überprüfung des Formats	1	Abschnitt B 2.3.7	Der Wert ' <code><WERT></code> ' des Datenfeldes <code><Feld.name></code> ' <code><TdsFeld.bezeichnung></code> ' ist kein gültiger <code><BasisTyp.name></code> -Wert (<code><BasisTyp.bezeichnung></code> <code><BasisTyp.format></code>).
Überprüfung der Feldlänge	2	Abschnitt B 2.3.7	Der Wert ' <code><WERT></code> ' des Datenfeldes <code><Feld.name></code> ' <code><TdsFeld.bezeichnung></code> ' überschreitet die zulässige Feldlänge <code><Feld.laenge></code> .
Sind in den Datenfeldern mit internen und externen	3	Abschnitt B 2.3.7	Ungültiger Schlüsselcode <code><Wert></code> des Schlüssels <code><Schluessel.name></code>

Beschreibung des Fehlers	Fehler-kode	Formale Fehlerbedingung	Fehlermeldung
Schlüsseln (Basis-typ SCHLUESSEL oder NUMSCHLUESSEL) gültige Schlüsselcodes eingetragen? ⁴³			im Datenfeld <Feld.name> '<TdsFeld.bezeichnung>'!
Überprüfung numerischer Wertebereiche	4	Abschnitt B 2.3.7	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 2.3.7: Attribut <TdsFeld.fkMussKann>	Das Datenfeld <Feld.name> '<TdsFeld.bezeichnung>' muss einen gültigen Wert enthalten.
Ist der Fall dem Spezifikationsjahr der Spezifikation zugeordnet? (Spezifikation <SPEZIFIKATIONSJAHR>)	6	(MODUL <> 'PCI' ODER (MODUL IN ('PCI'; 'DIAL')) UND FALLART = 2) UND AUFNDATUM <'01.01.<SPEZIFIKATIONSJAHR>' ODER AUFNDATUM '31.12.<SPEZIFIKATIONSJAHR>') ODER (MODUL = 'PCI' UND FALLART <> 2 UND (OPDATUM < '01.01.<SPEZIFIKATIONSJAHR>' ODER OPDATUM > '31.12.<SPEZIFIKATIONSJAHR>') ODER (MODUL = DIAL UND FALLART <> 2	Der Fall ist im Jahr <SPEZIFIKATIONSJAHR> nicht dokumentationspflichtig; Aufnahmedatum/OP-Datum = <Wert>

⁴³ 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
		UND (EBMDATUM < '01.01.<SPEZIFIKATIONSJAHR>' ODER EBMDATUM > '31.12.<SPEZIFIKATIONSJAHR>')	

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:

- Fehlerkode 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_<SPEZIFIKATIONSJAHR>_methodisch_DeQS.docx“
- „Vorlage_Sollstatistik_<SPEZIFIKATIONSJAHR>_Budget_DeQS.docx“
- „Vorlage_Sollstatistik_<SPEZIFIKATIONSJAHR>_SV_DeQS.docx“
- „Vorlage_Sollstatistik_<SPEZIFIKATIONSJAHR>_SV_KVD.docx“

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 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 Spezifikationsjahres dar.

2.1 Patientenidentifizierende Daten zur Follow-up-Erhebung

Hinweis:

Dieses Kapitel ist nicht relevant für das QS-Verfahren Ambulante Psychotherapie.

2.2 Datenfeldbeschreibung

Für jedes Modul existiert eine eigene Datenfeldbeschreibung. Sie spezifiziert alle auszufüllenden Datenfelder (Bogenfelder, auch Items genannt) und besteht aus mehreren Tabellen (Abbildung 11), die in den nachfolgenden Abschnitten erläutert werden.

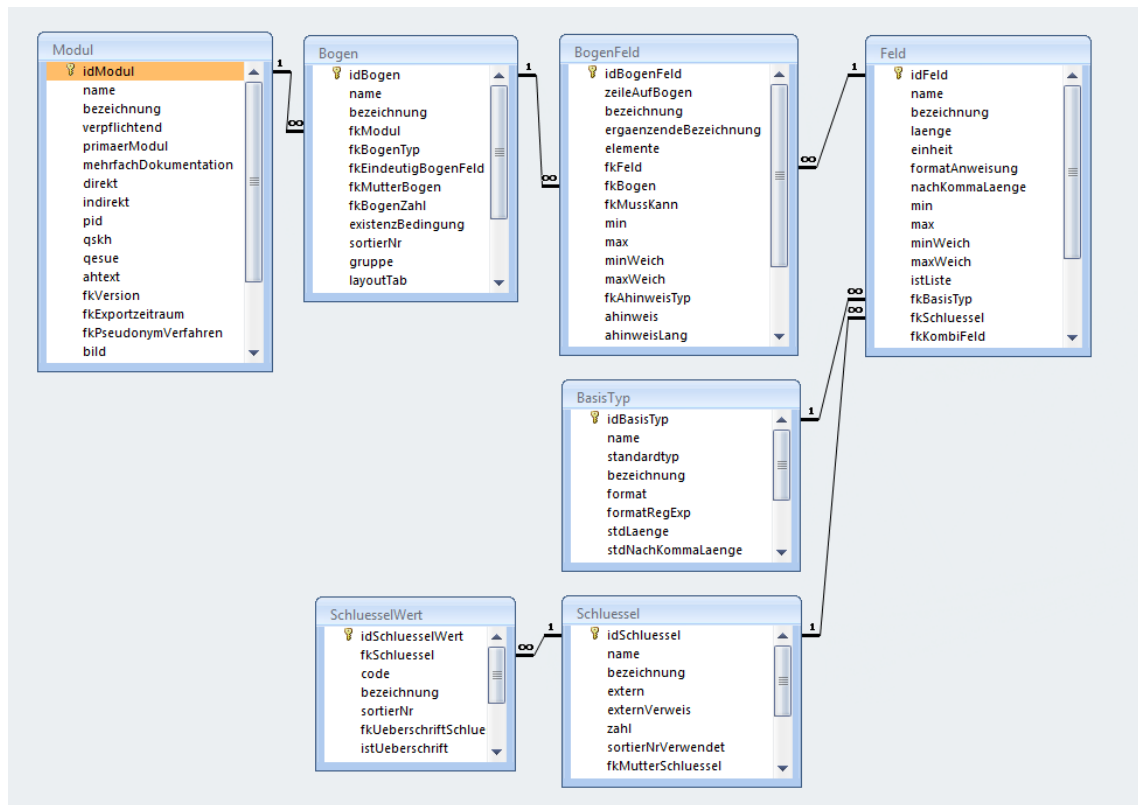


Abbildung 11: Tabellen und Relationen der Datenfelddescription (Beispiel Spezifikation 2017)

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 Datenfelddescription 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 Prozessablauf⁴⁴ eines Leistungserbringers einordnet.

2.2.1 Dokumentationsmodule (Datensätze)

Ein Modul der Spezifikation enthält die Datensatzdefinition von einem oder mehreren Verfahren. Abhängig von (inhaltlich oder organisatorisch) abzugrenzenden Bereichen kann ein Verfahren mehrere Module umfassen (Beispiel: APSY und PAPSY sind Module des QS-Verfahrens ambulante Psychotherapie). 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.

Tabelle 36: Struktur der Tabelle `Modul`

Feldname	Datentyp	Bemerkung
<code>idModul</code>	INTEGER	Primärschlüssel
<code>Name</code>	KURZER TEXT (32)	Eindeutiger technischer Name
<code>bezeichnung</code>	KURZER TEXT (255)	Erläuternde Bezeichnung
<code>verpflichtend</code>	BOOLEAN	Besteht für das Modul eine QS-Dokumentationsverpflichtung?
<code>primaerModul</code>	BOOLEAN	Ist das Modul ein Primärmodul?
<code>mehrfachDokumentation</code>	BOOLEAN	Ist ein mehrfaches Anlegen eines gleichartigen Datensatzes pro Fall zulässig (WAHR/FALSCH)?
<code>ahntext</code>	LANGER TEXT	Einleitender Text für den Ausfüllhinweis eines Moduls
<code>fkVersion</code>	INTEGER	Gültige Version des jeweiligen Moduls
<code>bild</code>	KURZER TEXT (20)	Modulspezifisches Bild

⁴⁴ Zum Beispiel in den Prozessablauf eines Krankenhauses.

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 Gebührenordnungspositionen (EBM-Kodes) zurück, die im Praxisinformationssystem verfügbar sind.

2.2.2 Teildatensätze

Hinweis:

Dieses Kapitel ist nicht relevant für das QS-Verfahren Ambulante Psychotherapie

2.2.3 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 Bezeichnung⁴⁵ 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 2.2.1), der datentechnische Typ (`BasisTyp`) charakterisiert das Format des Feldes.

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 Datenfeldname sogar innerhalb eines Moduls eindeutig sein, d.h. dass eine Abfrage mit dem Primärschlüsselpaar (`idModul`, `idFeld`) genau einen Primärschlüssel `idBogenFeld` liefert.

Tabelle 37: Struktur der Tabelle `BogenFeld`

Feldname	Datentyp	Bemerkung
<code>idBogenFeld</code>	INTEGER	Primärschlüssel
<code>zeileAufBogen</code>	DOUBLE	bestimmt die Reihenfolge von Datenfeldern im Dokumentationsbogen

⁴⁵ Gegebenenfalls im Kontext der Überschriften (Abschnitt B 2.3.4).

Feldname	Datentyp	Bemerkung
gliederungAufBogen	KURZER TEXT	Gliederungsnummer, die im Dokumentationsbogen angezeigt wird
bezeichnung	KURZER TEXT	Beschreibender Text zum Feld auf dem Dokumentationsbogen
ergaenzendeBezeichnung	KURZER TEXT	Optionale ergänzende Bezeichnung zu einem Bogenfeld.
elemente	INTEGER	Anzahl der Elemente bei Listenfeldern
fkFeld	INTEGER	Fremdschlüssel zu dem Teildatensatz und zu dem Feld, bilden zusammen die identifizierenden Merkmale
fkBogen	INTEGER	
fkMussKann	KURZER TEXT (1)	M oder K, Unterscheidung zwischen Muss- und Kann-Feldern
Min	DOUBLE	Harte Untergrenze des Wertebereichs eines numerischen Datenfeldes (modulspezifisch). Die Definition ist optional.
Max	DOUBLE	Harte Obergrenze des Wertebereichs eines numerischen Datenfeldes (modulspezifisch). Die Definition ist optional.
minWeich	DOUBLE	Weiche Untergrenze des Wertebereichs eines numerischen Datenfeldes (modulspezifisch). Die Definition ist optional.
maxWeich	DOUBLE	Weiche Obergrenze des Wertebereichs eines numerischen Datenfeldes (modulspezifisch). Die Definition ist optional.
ahinweis	KURZER TEXT (32)	Name des HTML-Ausfüllhinweises ohne Endung .htm (Abschnitt B 2.2.5)
fkICDListe	INTEGER	Zuordnung einer ICD-Liste
fkOPSListe	INTEGER	Zuordnung einer OPS-Liste

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 2.2.2).⁴⁶
- Kann-Felder (K) sind optionale Felder.
- Abhängige Muss-Felder (K) müssen nur unter bestimmten Bedingungen ausgefüllt werden. Wenn also logische Sachverhalte dem Ausfüllen von Kann-Feldern entgegenstehen, so dürfen

⁴⁶ In jedem Muss-Feld muss für jeden angelegten Teildatensatz einmal eine Angabe erfolgen.

sie nicht ausgefüllt werden. Diese Felder unterliegen Feldgruppenregeln und verfügen wie optionale Felder über den Attributwert `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.

Verpflichtend zu dokumentierende ICD- und OPS-Kodes

Über die Attribute `fkICDListe` und `fkOPSListe` können Code-Listen mit verpflichtend zu dokumentierenden Codes referenziert werden. Dies ist von Bedeutung, wenn neben der Dokumentation von auslösenden Codes die Angabe weiterer Codes erwartet wird. Beispielsweise gibt es im Verfahren Cholezystektomie die Anforderung, dass in den Feldern zu Operationen und Diagnosen bestimmte Codes, über die Einschlusscodes hinaus, aus definierten Listen übernommen werden müssen. Hierzu können die Einschlusslisten des QS-Filters mit der Liste der zusätzlichen Codes zusammengeführt werden und die Gesamtliste in der Tabelle `BogenFeld` referenziert werden. (siehe Abschnitt B 2.4.3).

Felder – ein erster Schritt zur Prozess- und Datenintegration

Die Tabelle `Feld` (Tabelle 38) 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 2.2.1). 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`⁴⁷ 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. Bei der Wahl des technischen Feldnamens sind beispielsweise folgende Punkte zu berücksichtigen:

⁴⁷ Das Attribut `bezeichnung` ist ein Standardtext für das gleichnamige Attribut der Tabelle `BogenFeld`. Im Eingabeformular wird die Bezeichnung aus der Tabelle `BogenFeld` angezeigt.

- Der technische Name eines Feldes muss eindeutig sein
- Bei der Erstellung des technischen Feldnamens ist im Idealfall eine Zeichenlänge von circa 1 bis 15 Zeichen zu nutzen. Eine Zeichenlänge von 25 sollte im Idealfall nicht überschritten werden.
- Sonderzeichen wie z.B. Unterstriche sollten nicht verwendet werden.
- Der Feldname wird in Großbuchstaben geschrieben.
- Der technische Name ist so zu wählen, dass dieser den abzufragenden Sachverhalt zwar grob umfasst, jedoch nicht so konkret, dass eine modulübergreifende Verwendung des Feldes nicht mehr möglich ist.
- Nachträgliche Anpassungen der Bezeichnung des technischen Feldes bei Änderungen der Bogenfeldbezeichnung werden i.d.R. nicht vorgenommen.

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.⁴⁸ Die Anzahl der Elemente des von einem Feld abgeleiteten Bogenfelds wird über das Attribut `elemente` der Tabelle `BogenFeld` gesteuert.

Beispiel:

Das Feld `ENTLDIAG` (Aufnahmediagnosen) ist als Liste definiert.

Im Modul `APSY` enthält das entsprechende Bogenfeld 10 Elemente (`BogenFeld.elemente`), im Modul `16/1` hat das Bogenfeld lediglich 5 Elemente.

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 38: Struktur der Tabelle `Feld`

Feldname	Datentyp	Bemerkung
<code>idFeld</code>	INTEGER	Primärschlüssel
<code>name</code>	KURZER TEXT	Technischer Name
<code>bezeichnung</code>	KURZER 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>)

⁴⁸ Man beachte die Besonderheiten der Listenfelder beim Datenexport und in der Syntax der Plausibilitätsregeln.

Feldname	Datentyp	Bemerkung
laenge	INTEGER	Anzahl der Zeichen in der Feldeingabemaske, enthält beim Typ ZAHL auch das Komma, bei SCHLUESSEL die Trennzeichen
einheit	KURZER TEXT (50)	Einheit des Felds (z. B. mm, Stunden)
formatAnweisung	KURZER TEXT	Regulärer Ausdruck für die Formatprüfung (z. B. [0-9]{9})
funktion	KURZER TEXT	Formel zur Generierung des Feldinhaltes, z. B. durch Aufruf einer Syntaxfunktion
nachKommaLaenge	INTEGER	Anzahl der Nachkommastellen in der Feldeingabemaske (muss kleiner als laenge 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.
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 istListe = WAHR, 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
strukturParameter	BOOLEAN	Markierung von Strukturparametern, die durch die Software automatisch aus dem PVS in den Dokumentationsbogen übernommen werden können

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 2.3.2).

Das identifizierende Merkmal eines Basistyps ist sein technischer Name (Attribut `name`).

Tabelle 39: Struktur der Tabelle *Basistyp*

Feldname	Datentyp	Bemerkung
<code>idBasistyp</code>	INTEGER	Primärschlüssel
<code>Name</code>	KURZER TEXT	Technischer Name (muss eindeutig sein)
<code>standardtyp</code>	KURZER TEXT	Entsprechender Standarddatentyp
<code>bezeichnung</code>	KURZER TEXT	Beschreibender Text
<code>Format</code>	KURZER TEXT	Formatdefinition, z. B. <code>TT.MM.JJJJ</code> beim Basistyp <code>Datum</code>
<code>formatRegExp</code>	KURZER 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 (Kodesystem) ist sein technischer Name. Die meisten Schlüsselcodes sind in der Tabelle *Schlüsselwert* (Tabelle 41) definiert.

Tabelle 40: Struktur der Tabelle *Schluesse**l*

Feldname	Datentyp	Bemerkung
idSchluessel	INTEGER	Primärschlüssel
name	KURZER TEXT	Technischer Name (muss eindeutig sein)
bezeichnung	KURZER TEXT	Beschreibender Text
extern	BOOLEAN	Zeigt an, ob der Schlüssel in der Tabelle Schlues- sel (= FALSCH) oder in einer externen Tabelle ge- speichert (= WAHR) ist.
externVerweis	KURZER TEXT	Verweis auf die Quelle des externen Schlüssels
zahl	BOOLEAN	Wenn WAHR, sind die Werte im Attribut code der zu- gehörigen Schlüsselwerte als ganze Zahl kodiert, an- sonsten als Zeichenkette.
sortierNrVerwendet	BOOLEAN	Flag, das anzeigt, ob für die Reihenfolge das Attribut sortierNr der Tabelle SchluesseWert her- angezogen wird.
fkMutterSchluessel	INTEGER	Referenz auf einen übergeordneten Schlüssel. Bei- spielsweise enthält der Schlüssel MaSarkome ausschließlich Codes des Schlüssels ICDO3Mamma. Abgeleitete Schlüssel enthalten in der Regel keine Bezeichnungen (Datenbanktabelle SchluesseWert), da diese bereits im „Mutter- schlüssel“ definiert sind.

Schlüsselcodes können auf zwei Arten interpretiert werden: Wenn das Attribut `zahl` ge-
setzt ist, so werden die Codes als ganze Zahl gedeutet, ansonsten werden sie als Zeichenket-
ten interpretiert. In der Syntax der Plausibilitätsregeln werden die letztgenannten Codes in
einfache Hochkommata gesetzt (Abschnitt B 2.3.2).

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.bfarm.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.bfarm.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 `SchluessselWert` enthalten. Spätere Schlüsseländerungen bzw. -fortschreibungen werden vom IQTIG zeitnah übernommen.

Schlüsselwerte

Tabelle 41 gibt einen Überblick über die Datenbanktabelle `SchluessselWert`, in der die Codes und Bezeichnungen der Schlüssel hinterlegt sind. Identifizierendes Merkmal ist hier eine Kombination der Spalten `fkSchluesssel` und `code`. Das bedeutet, dass jeder Schlüsselcode innerhalb eines Schlüssels nur einmal vorkommen darf.

Tabelle 41: Struktur der Tabelle *SchlüsselWert*

Feldname	Datentyp	Bemerkung
idSchlüsselWert	INTEGER	Primärschlüssel
fkSchlüssel	INTEGER	Fremdschlüssel zur Tabelle Schlüssel
code	KURZER TEXT (50)	Schlüsselcode (entweder numerisch oder alphanumerisch kodiert)
bezeichnung	KURZER TEXT	Textliche Definition des Schlüsselwertes
sortierNr	INTEGER	Optionale Angabe zur Reihenfolge der Schlüsselwerte: Wenn belegt, so ist diese Reihenfolge bei der Anzeige in der Erfassungssoftware einzuhalten.

Das Attribut `code` der Tabelle `SchlüsselWert` 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 2.3.2 und B 2.3.6) 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 `SchlüsselWert.code`) eines Schlüssels ist eine Sortierung definiert. Die Art der Sortierung wird über die Attribute `zahl` und `sortierNrVerwendet` der Tabelle `Schlüssel` festgelegt.
- Numerische Sortierung: Wenn `sortierNrVerwendet = FALSCH` und `zahl = WAHR`, so sind die Codes nach der Spalte `code` der Tabelle `Schlüssel` numerisch zu sortieren.
- Alphanumerische Sortierung: Wenn `sortierNrVerwendet = FALSCH` und `zahl = FALSCH`, so sind die Codes nach der Spalte `code` der Tabelle `Schlüssel` alphanumerisch zu sortieren.
- Spezielle Sortierung: Wenn `sortierNrVerwendet = WAHR`, so sind die Codes nach den Werten in der Spalte `sortierNr` der Tabelle `Schlüssel` numerisch zu sortieren.

Beispiel:

Das Datenfeld `GRUBEE` des Datensatzes `APSY` besitzt den Schlüssel `GrundBeend`, für den die spezielle Sortierung (`sortierNrVerwendet = WAHR`) definiert ist.

Tabelle 42: Schlüssel mit spezieller Sortierung (pTMamma)

Code	sortierNr
1	1
3	2
7	3
5	4
6	5
8	6

2.2.4 Überschriften

Die Überschriften der Dokumentationsbögen in der Spezifikation sind in der Tabelle *Abschnitt* zu finden.

Tabelle 43: Struktur der Tabelle *Abschnitt*

Feldname	Datentyp	Bemerkung
idAbschnitt	INTEGER	Primärschlüssel
bezeichnung	KURZER 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.

2.2.5 Ausfüllhinweise

Die Ausfüllhinweise zu den Datenfeldern sind in einem separaten ZIP-Archiv enthalten, das nach dem Benennungsschema für Spezifikationskomponenten bezeichnet wird (TechDok_Benennungsschema). 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 (einrichtunginterne) 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 44: Arten von Hinweistypen (Spezifikation 2018)

<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 HEP, KEP und PCI.
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 DEK.
Speziell	Soll es für verschiedene Datenfelder der Tabelle <code>Feld</code> einen gemeinsamen Ausfüllhinweis geben, kann dieser als speziell deklariert werden. Der	<code>NICHTGEMESSENVE.htm</code>

	Attributwert <code>ahinweis</code> definiert den Namen des Ausfüllhinweises.	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.

2.3 Plausibilitätsprüfungen

Es wird zwischen zwei Arten von Plausibilitätsprüfungen unterschieden, die in Tabelle `RegelTyp` definiert sind:

- harte Prüfungen
- weiche Prüfungen in der QS-Dokumentationssoftware

Tabelle 45: Tabelle `RegelTyp`

<code>idRegelTyp</code>	<code>bezeichnung</code>
H	Hart
W	Weich

Weiterhin wird zwischen sogenannten **Einzelregeln** (Abschnitt B 2.3.4) und **Feldgruppenregeln** (Abschnitt B 2.3.6) unterschieden.

2.3.1 Die Regeltabelle

Die Bedingungen für unplausible Angaben⁴⁹ sind in der Tabelle `Regeln` abgelegt. Die hier beschriebenen Prüfungen sind in der Spezifikationsdatenbank für QS-Dokumentation hinterlegt. Die Syntax ist in Abschnitt B 2.3.4 beschrieben. Die Bedingungen sind möglichst kurz gefasst (Vermeidung von durch `ODER` verknüpften Teilbedingungen). Jede Bedingung kommt nur einmal innerhalb eines Moduls vor.

Tabelle 46: 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>

⁴⁹ Eine Plausibilitätsregel müsste eigentlich „Unplausibilitätsregel“ heißen, weil sie unplausible Zustände beschreibt, die zu Fehlermeldungen führen.

Feldname	Datentyp	Bemerkung
bedingung	LANGER TEXT	Entsprechend der Syntax definierte Regeln
meldung	LANGER TEXT	Fehlermeldung: Diese Texte sind bei Regeln mit Bezug zu Feldgruppen automatisch generiert.
alternativMeldung	LANGER TEXT	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	KURZER TEXT (1)	Fremdschlüssel zur Tabelle <code>RegelTyp</code> : Die Regeltypen sind die in Abschnitt B 2.3 beschriebenen Arten der Plausibilitätsprüfungen: H oder W
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 47) 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 47: 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 48: Struktur der Tabelle *MehrfachRegel*

Feldname	Datentyp	Bemerkung
idMehrfachRegel	INTEGER	Primärschlüssel
bedingung	LANGER TEXT	veraltet: entspricht QSDOK.regeln.bedingung(fkMehrfachregel)
meldung	LANGER TEXT	<u>veraltet: entspricht QSDOK.regeln.meldung(fkMehrfachregel)</u>
ersatzBedingung	LANGER TEXT	Ersatzregel für den pseudonymisierten Datensatz
fkRegelTyp	KURZER TEXT (1)	Fremdschlüssel zur Tabelle RegelTyp: Die Regeltypen sind die in Abschnitt A 2.3 beschriebenen Arten der Plausibilitätsprüfungen: H, W oder D

Beispiel:

Die Datenfelder der Regel 19865 (`GEBDATUM > aktuellesDatum()`) werden nicht an die Datenannahmestellen übermittelt. Bei der Entgegennahme ist die in der Tabelle *MehrfachRegel* definierte Ersatzregel (`idMehrfachRegel = 160`) anzuwenden: `gebjahr>jahreswert(DokAbschlDat)`.

Weitere Regeln

Weitere feldübergreifende Regeln sind die in Abschnitt B 2.2.2 beschriebenen Existenzbedingungen für das Anlegen von abhängigen Teildatensätzen (Attribut `existenzBedingung` in Tabelle *Bogen*).

2.3.2 Regelsyntax

Bedingungen sind in den Tabellen *Regeln*, *MehrfachRegel* und *Bogen* definiert. Die den Bedingungen zugrundeliegende 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 49 aufgelistet.

Tabelle 49: Basistypen der Datenfelder in den Plausibilitätsregeln

Basistyp	Bezeichnung	Beispiele (Literele)
BOOL	Boolesche Variable	WAHR, FALSCH
TEXT	Zeichenkette (String)	"Spezifikation"

GANZEZAHL ⁵⁰	... -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 Bogenfelder⁵¹ enthalten sein. Bei der Evaluierung von Regeln werden die aktuellen Werte der referenzierten Bogenfelder eingesetzt. Kann-Bogenfelder können auch unausgefüllt sein, also den Wert LEER haben.

⁵⁰ Beim Typ GANZEZAHL sind auch negative ganze Zahlen erlaubt.

⁵¹ 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 `EBM`, welches im Modul `APSY` als Bogenfeld vorkommt, verbirgt sich eine Liste mit 3 Elementen, die nachfolgend als Variable in einer Regel angesprochen wird:

```
EBM EINSNIGHTIN (LEER) UND EBM KEINSIN APSY_EBM
```

Literale

Alphanumerische Literale (z. B. `SCHLUESSEL`) werden von einfachen Hochkommata eingeschlossen, während Zeichenketten vom Datentyp `TEXT` in Anführungszeichen gesetzt werden müssen.⁵²

Dies gilt nicht für numerische Literale (`GANZEZAHL`, `ZAHL`, `NUMSCHLUESSEL` und `JAHRDATUM`) 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 Diagnosecodes (ICD-GM-10) werden als Variable angesprochen, deren Namen einem festen Namensschema gehorchen. Diese Listen werden in separaten Tabellen definiert, die

⁵² Beim Export entfallen die begrenzenden Zeichen.

den Variablennamen⁵³ (z. B. GYN_OPS) und die darin enthaltenen Prozedur- und Diagnosecodes beinhalten (Abschnitt B 2.4).

Beispiel:

In der Regel 19860 aus dem Modul APSY wird die EBM-Liste APSY_EBM verwendet:

```
EBM EINSNICHTIN (LEER) UND EBM KEINSIN APSY_EBM
```

Operatoren

Tabelle 50 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 50 hat jeder einzelne Operator eine Präzedenzstufe (höchste Präzedenzstufe ist 0). Operatoren, die die gleiche Stufe haben, werden nach den Regeln der Assoziativität aufgelöst.

Tabelle 50: 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“
	links	>=	Vergleichsoperator „größer gleich“
4	links	=	Vergleichsoperator „gleich“

⁵³ Der Variablenname ist synonym mit dem Listennamen (z.B. OPSListe.name) der Prozedur- bzw. Diagnoseliste.

	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 (ID 19870) prüft, ob kein Element des Listenfeldes `EBM` (3 Elemente) einen bestimmten Code besitzt:

```
EBM EINSNICHTIN (LEER) UND EBM KEINSIN APSY_EBM
```

Mit dieser Regel wird zum einen überprüft, dass mindestens eine Angabe in der `EBM`-Liste vorliegt (`EBM EINSNICHTIN (LEER)`). Zum anderen wird die `EBM`-Liste gegen die Einschlussliste geprüft, damit zumindest ein Einschluss-Kode angegeben ist.

Plausibilitätsprüfungen mit OPS- und ICD-Listen

Die OPS- und ICD-Listen enthalten ausschließlich Normkodes. Die vom Leistungserbringer dokumentierten Codes enthalten ggf. auch Zusatzkennzeichen (Bsp.: Seitenlokalisierung). Bei der Evaluation der Regeln werden die dokumentierten Zusatzkennzeichen ignoriert (Abschnitt B 2.4).

Beispiel:

Die `EBM`-Liste `APSY_EBM` enthält unter anderem den Code 35411.

Die Evaluation der Regel `EBM EINSNICHTIN (LEER) UND EBM KEINSIN APSY_EBM` führt auch dann zu einem positiven Ergebnis, wenn `EBM = ('35411';LEER;LEER;LEER)`

2.3.3 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 0).

Beispiel:

Das Ersatzfeld `entlquartal` wird mithilfe der Syntaxfunktion `quartal` berechnet:

```
entlquartal = quartal (ENTLDATUM)
```

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:⁵⁴

- 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.

```
{
```

⁵⁴ Der Pseudocode erhebt nicht den Anspruch auf formale Korrektheit.

```
...
}
```

- Innerhalb einer Funktion sind die Argumentvariablen verfügbar.
- Eine Variable, die den gleichen Namen wie die Funktion hat, muss am Ende mit `return` zurückgegeben werden.

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 (VERSICHERTENIDNEU; ' [A-Z] [0-9] {9} ')
```

[A-Z] : Die (zehnstellige) Zeichenkette beginnt mit einem Zeichen, das nur einen Wert A-Z enthalten darf.

[0-9] : Die darauffolgenden (neun) Werte dürfen nur die Zahlen 0-9 enthalten.

{9} : der Wert (hinter dem Buchstaben) muss genau 9-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>

2.3.4 Syntaxvariablen

Der Eingangsdatensatz bildet die medizinische Routedokumentation ab, die in jedem Praxisverwaltungssystem (PVS) 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.

Jeder der in der Tabelle `SyntaxVariable` definierten Variablen ist über den Wert des Attributes `SyntaxVariable.fkTdsFeld` ein Feld des Risikostatistik- bzw. QS-Filter-Eingangsdatensatzes zugeordnet.

Jedes dieser Felder besitzt somit einen Basistyp.

Die in den Bedingungen 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 Feld-

name 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`).

2.3.5 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 2.3.6 beschriebenen Feldgruppenregeln 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 2.3.7 beschriebenen Prüfungen des Formates, der Feldlänge, der Wertebereiche, Prüfungen von Schlüsselcodes 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 Angaben⁵⁵.

2.3.6 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 Antworten⁵⁶ eines jeden Datenfeldes werden in zwei Gruppen aufgeteilt. Die erste Gruppe ist die Menge der positiven, die zweite Gruppe die Menge der negativen Antworten.⁵⁷

Typische positive Antworten sind beispielsweise:

```
Feld <> LEER oder Feld IN (2;3)
```

⁵⁵ Eine Plausibilitätsregel müsste eigentlich „Unplausibilitätsregel“ heißen, weil sie unplausible Zustände beschreibt, die zu Fehlermeldungen führen.

⁵⁶ Die Antworten eines Datenfeldes umfassen hier neben möglichen Werten (z.B. Schlüsselwerten) oder Wertemengen auch die Kategorie „nicht ausgefüllt“ (LEER).

⁵⁷ Die negativen Antworten sind abhängig von der definierten Bedingung eines Feldes in der entsprechenden Feldgruppe.

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. Die folgende Tabelle gibt einen Überblick über die Typen von Feldgruppen. Der aktuelle Stand findet sich in der Tabelle `FeldGruppenTyp` der Spezifikation.

Tabelle 51: 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
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.⁵⁸

Struktur der Tabellen `FeldGruppe` und `FeldgruppeFelder`

Die Feldgruppen sind in den Tabellen `FeldGruppe` und `FeldgruppeFelder` definiert. In der Tabelle `FeldGruppe` (Tabelle 52) sind Name, Typ und die Zuordnung zu einem Modul definiert. Die Verknüpfungstabelle `FeldgruppeFelder` (Tabelle 53) definiert die abhängigen Bogenfelder. Zusätzlich wird hier festgelegt, welche Bogenfelder der Feldgruppe als Filterfeld dienen.

Tabelle 52: Struktur der Tabelle `FeldGruppe`

Feldname	Datentyp	Bemerkung
<code>idFeldGruppe</code>	INTEGER	Primärschlüssel
<code>name</code>	KURZER TEXT (64)	Technischer Name der Feldgruppe
<code>fkModul</code>	INTEGER	Obligatorischer Fremdschlüssel zu einem Modul
<code>fkFeldgruppenTyp</code>	INTEGER	Obligatorischer Fremdschlüssel zu einem Feldgruppentyp
<code>hinweis</code>	KURZER 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.
<code>fkFilterFeldTyp</code>	CHAR (1)	Attribut wird bei Feldgruppen mit mehreren Filterfeldern gesetzt: O = Oder-Verknüpfung der positiven Filterbedingungen U = Und-Verknüpfung der positiven Filterbedingungen
<code>fkRegelTyp</code>	CHAR (1)	Fremdschlüssel zur Tabelle <code>RegelTyp</code> : Die Regeltypen sind die in Abschnitt A 2.3 beschriebenen Arten der Plausibilitätsprüfungen: H, W oder D

⁵⁸ Die Funktion `verkettent` 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
		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 S. 167, Layout-Feldgruppen)

Tabelle 53: Struktur der Tabelle *FeldgruppeFelder*

Feldname	Datentyp	Bemerkung
idFeldgruppeFelder	INTEGER	Primärschlüssel
fkFeldGruppe	INTEGER	Obligatorischer Fremdschlüssel zur Feldgruppe
fkBogenFeld	INTEGER	Obligatorischer Fremdschlüssel zum Bogenfeld
bedingung	KURZER TEXT	Positive Bedingung für das jeweilige Bogenfeld
istFilter	BOOL	Legt fest, ob das jeweilige Bogenfeld ein Filterfeld ist
bezeichnungSchluesselListe	KURZER 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 1.1.1) 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 50 verwendet werden. Die auf der rechten Seite erlaubten Operanden sind nachfolgend aufgelistet:

- Literale (Tabelle 49)
- LEER
Kodelisten, in denen auch die Codes eines Schlüssels referenziert werden können;
Beispiel: (1;2;3) oder ENTLDIAG
- ICD-Listen oder OPS-Listen (z. B. APSY_ICD_EX)

**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 54: 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)

$$n(F) \text{ UND } p(A_i) \quad i=1, \dots, n$$

$$p(F) \text{ UND } n(A_1) \text{ UND } n(A_2) \text{ UND } \dots \text{ UND } n(A_n)$$

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)

$$n(F) \text{ UND } p(A_i) \quad i=1, \dots, n$$

$$p(F) \text{ UND } n(A_i) \quad i=1, \dots, n$$

Insgesamt sind $2n$ Einzelregeln mit der Feldgruppe verknüpft.

- Regeln der Feldgruppe „Einfachauswahl mit Filter“ (EF_FILTER)

$$n(F) \text{ UND } p(A_i) \quad i=1, \dots, n$$

$$p(F) \text{ UND } n(A_1) \text{ UND } n(A_2) \text{ UND } \dots \text{ UND } n(A_n)$$

$$p(F) \text{ UND } p(A_j) \text{ UND } p(A_i) \quad \text{für alle unterschiedlichen } i, j = 1, \dots, n$$

Insgesamt sind $n(n+1)/2 + 1$ Einzelregeln mit der Feldgruppe verknüpft.

- Regeln der Feldgruppe „Optionale Einfachauswahl mit Filter“ (EF_OPTIONAL_FILTER)

$$n(F) \text{ UND } p(A_i) \quad i=1, \dots, n$$

$$p(F) \text{ UND } p(A_j) \text{ UND } p(A_i) \quad \text{für alle unterschiedlichen } i, j = 1, \dots, n$$

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:

$$p(F) \text{ UND } n(A1) \text{ UND } n(A2) \text{ UND } \dots \text{ UND } n(An)$$

$$p(F) \text{ UND } p(A_j) \text{ UND } p(A_i) \text{ für alle unterschiedlichen } i, j = 1, \dots, n$$
Feldgruppen ohne Filter

- Regeln der Feldgruppe „Einfachauswahl“ (`EF`)

$$n(A1) \text{ UND } n(A2) \text{ UND } \dots \text{ UND } n(An)$$

$$p(A_j) \text{ UND } p(A_i) \text{ für alle unterschiedlichen } i, j = 1, \dots, n$$

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 $F1, F2, \dots, Fn$ mit den positiven bzw. negativen Bedingungen $p(Fj)$ bzw. $n(Fj)$. Für alle Filterfelder wird eine positive Bedingung $p(F1, \dots, Fn)$ und eine negative Bedingung $n(F1, \dots, Fn)$ gebildet. Diese modifizierten Filterbedingungen ersetzen die im Abschnitt B 2.3.6 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:

$$p(F1, \dots, Fn) = p(F1) \text{ ODER } p(F2) \text{ ODER } \dots \text{ ODER } p(Fn)$$

(`ODER`-Verknüpfung)

$$p(F1, \dots, Fn) = p(F1) \text{ UND } p(F2) \text{ UND } \dots \text{ UND } p(Fn)$$

(`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 12 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 ☐ 0 = nein 1 = ja
wenn Operation(en) und Prozeduren während des aktuellen stationären Aufenthaltes = ja	
74>	OP oder Therapie einer ROP (Frühgeborenenretinopathie) ☐ 0 = nein 1 = ja, Lasertherapie oder Kryotherapie 2 = ja, intravitreale Anti-VEGF-Therapie 3 = ja, sonstige
75>	OP einer NEK (nekrotisierende Enterokolitis) ☐ 0 = nein 1 = ja
76>	Indikation(en) zu(r) weiteren Operation(en) http://www.dimdi.de 1. ☐☐☐ . ☐☐ 2. ☐☐☐ . ☐☐ 3. ☐☐☐ . ☐☐ 4. ☐☐☐ . ☐☐ 5. ☐☐☐ . ☐☐

Abbildung 12: 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.

2.3.7 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 2.3.4 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 0): 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 Länge⁵⁹ (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 `<Schluessel.name>` (`<Schluessel.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 kein zulässiger Code des Schlüssels `<Schluessel.name>` (`<Schluessel.bezeichnung>`).

⁵⁹ Wenn bei einem Ersatz die Länge nicht spezifiziert ist, so entfällt die Prüfung.

Standardisierter Fehlertext für Muss-Fehler eines Zusatzfeldes

Das Zusatzfeld `<Modul.name>:<Bogen.name>:<ZusatzFeld.name> ' <ZusatzFeld.bezeichnung>` ist kein zulässiger Code des Schlüssels `<Schluessel.name> (<Schluessel.bezeichnung>).`

Beispiel (Bogenfeld):

Der Wert `'xyz'` des Datenfeldes `HEP:PROZ:OPSCHLUESSEL 'Prozedur(en)'` (Zeile 29) ist kein zulässiger Code des Schlüssels `OPS301`.

Besonderheiten bei externen Schlüsseln

- Bei externen BfArM-Schlüsseln (ICD-10-GM oder OPS) sind die jeweils gültigen amtlichen Kataloge zu verwenden. Alle Codes eines Behandlungsfalles müssen in derjenigen Katalogversion dokumentiert sein.⁶⁰
- 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 2.4.1).

Überprüfung numerischer Wertebereiche

Bei numerischen Datenfeldern (Typ `ZAHL` oder `GANZEZAHL`) 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.

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

⁶⁰ Analog zur Regelung zur Abgrenzung von Spezifikationsjahren und Datensatzformaten in Abschnitt B 2.7.3.

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>'.

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 2.2.3) 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.

2.3.8 Verfahren für die Evaluation von Regeln

Jedes Bogenfeld – jedes Listefeld – darf keine Regel aus der Tabelle QSDOK.Regeln verletzen. Nach der Dokumentation muss jedes Exportfeld die QSDOK.Regeln mit `gueltigNachExport` sowie die referenzierten Mehrfachregeln einhalten.

Grundsätzlich muss jede gem. Abschnitt B 2.3.2 formulierte Regel evaluiert (ausgeführt) werden, wenn keine der folgenden Bedingungen zutrifft:

1. Mindestens ein referenziertes Bogenfeld ist auf einem nicht vorhandenen Teildatensatz.
2. Für mindestens ein referenziertes Bogenfeld⁶¹ schlägt eine harte Feldprüfung (Abschnitt B 2.3.7) fehl.⁶²
3. 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.
 - Jeder Operation auf einen Wert `<> LEER` ist eine `ODER`-Verknüpfte Prüfung auf `LEER` direkt vorgeschaltet (`Feld = LEER ODER Feld Operator Operand`).
4. Eine Funktion der Regel hat das Ergebnis `LEER` und wird in der Regel nicht explizit mit `<>` oder `=` auf `LEER` geprüft. (Hinweis: alle Argumente der Funktionen müssen dennoch/immer ungleich `LEER` sein.)

⁶¹ Die Verbindung zwischen Regeln und Bogenfeldern geschieht über die Tabelle `RegelFelder`, siehe Regelfelder (Bogenfelder einer Regel), Abschnitt B 2.4.1.

⁶² Erst bei Fehlerfreiheit der feldbezogenen Prüfungen werden die feldübergreifenden Prüfungen durchgeführt.

Abbildung 13: Algorithmus zur Evaluation von Plausibilitätsregeln

Tabelle 55: Beispiele für die Evaluation von Regeln

Beispiel		Erläuterung	Evaluation
HTX	MECHKREISLUNTER STUETZ <> 0 UND abstKreislauf 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

2.4 Listen von Schlüsselkodes (ICD-10-GM)

In der Spezifikation sind Listen von ICD-10-GM-Kodes⁶³ in separaten Tabellen definiert. Jede Liste hat einen technischen Namen (z. B. `APSY_ICD`) und eine erläuternde Bezeichnung (z. B. `Ein-schlussdiagnosen ambulante Psychotherapie`). Die Listen sind in insgesamt vier Tabellen der Spezifikationsdatenbank definiert. Die technischen Namen und erläuternden Bezeichnungen sind in der Tabelle `ICDListe` definiert. Die Codes finden sich in der Tabelle `ICD-Wert`.



Achtung

Einige Listen der Tabellen `ICDListe` sind als QS-Filter-Listen deklariert (Attribut `qsFilter`). Die Definitionen der QS-Filter-Listen sind in der für das Spezifikationsjahr gültigen Spezifikation für QS-Filter zu finden. Dort existieren die gleichen Tabellen `ICDListe` und `ICD-Wert` mit den entsprechenden Listendefinitionen.

⁶³ Die aktuell gültigen Kataloge sind über das BfArM (<http://www.bfarm.de>) zu beziehen.

2.4.1 OPS-Listen

Hinweis:

Dieser Abschnitt ist nicht relevant für das QS-Verfahren Ambulante Psychotherapie

2.4.2 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 BfArM:

2.4.3 Relationstabellen ICD- Listen

Über die Tabelle `ICDRelation` können Code-Listen zu Gesamtlisten zusammengeführt werden, um diese z. B. für Plausibilitätsprüfungen zu nutzen oder um sie einem Bogenfeld zuzuordnen. Beispielsweise gibt es im Verfahren Cholezystektomie die Anforderung, dass in den Feldern zu Operationen und Diagnosen bestimmte Codes, über die Einschlusscodes hinaus, aus definierten Listen übernommen werden müssen. Hierzu können die Einschlusslisten des QS-Filters mit der Liste der zusätzlichen Codes zusammengeführt werden und die Gesamtliste in der Tabelle `Bogenfeld` referenziert werden. (siehe Abschnitt B 2.2.3).

Beispiel:

Die Liste `APSY_ICD_GESAMT` setzt sich zusammen aus den Einzellisten `APSY_ICD`, `APSY_ICD_EX` und `APSY_ICD_ZUS`

In der Tabelle `ICDWert` werden die entsprechenden Codes nur für die Teillisten gepflegt. Die Werte der Gesamtlisten ergeben sich aus den Werten der Teillisten und werden in der Tabelle `ICDWert` nicht zusätzlich gepflegt.

2.5 Exportfeldbeschreibung

Neben der Datenfeldbeschreibung (Abschnitt B 2.2) 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 2.2.1) und Exportmodulen unterschieden.

2.5.1 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 Exportmodulen (Tabelle `Exportmodul`) unterschieden.

Tabelle 56: Struktur der Tabelle `ExportModul`

Feldname	Datentyp	Bemerkung
<code>idExportModul</code>	INTEGER	Primärschlüssel
<code>fkModul</code>	INTEGER	Bezug zum Dokumentationsmodul
<code>name</code>	KURZER TEXT	Technischer Name (muss eindeutig sein)
<code>bezeichnung</code>	KURZER TEXT	Bezeichnung des Exportmoduls
<code>existenzBedingung</code>	KURZER TEXT	Definiert, unter welcher Bedingung das Modul in ein definiertes Exportmodul transformiert wird.
<code>type_QS_data</code>	KURZER TEXT	Datentyp im XML-Schema
<code>ersatzBedingungMDS</code>	KURZER TEXT	Definiert, unter welcher Bedingung das Modul MDS in ein definiertes Exportmodul transformiert wird.
<code>type_QS_data_mds</code>	KURZER TEXT	Datentyp im XML-Schema
<code>fkExportzeit- raumEntlSJ</code>	INTEGER	Definiert den Exportzeitraum für ein Exportmodul bei Entlassung im Spezifikationsjahr. Die hinterlegten Zeitpunkte sind in der Tabelle <code>Exportzeitraum</code> definiert. <code>Exportzeitraum.exportBis</code> = jährliche Datenlieferfrist inkl. Korrekturfrist <code>Exportzeitraum.exportBisM12</code> = jährliche Datenlieferfrist ohne Korrekturfrist Bei Modulen zur Patientenbefragung gelten ausschließlich die monatlichen Datenlieferfristen.
<code>fkExportzeit- raumEntlSJ1</code>	INTEGER	Definiert den Exportzeitraum für ein Exportmodul bei Entlassung im Spezifikationsjahr + 1 (Langlieger). Die hinterlegten Zeitpunkte sind in der Tabelle <code>Exportzeitraum</code> definiert. <code>Exportzeitraum.exportBis</code> = jährliche Datenlieferfrist inkl. Korrekturfrist <code>Exportzeitraum.exportBisM12</code> = jährliche Datenlieferfrist ohne Korrekturfrist

Feldname	Datentyp	Bemerkung
		Bei Modulen zur Patientenbefragung gelten ausschließlich die monatlichen Datenlieferfristen.
pid	BOOLEAN	Handelt es sich um ein Modul zur Follow-up-Erhebung? ⁶⁴ (Abschnitt B 2.1)
fkPseudonymVerfahren	INTEGER	Pseudonymisierung von FU-Verfahren ⁶⁵
vpb	BOOLEAN	Dient der zur Kennzeichnung des Datenflusses an die Versendestelle für die Patientenbefragung.
fkRichtlinie	INTEGER	Verweis auf die für das Exportmodul gültige Richtlinie

Beispiel:

Das Modul `APSY` gliedert sich nach Abschluss der Dokumentation bei Erfüllen der Existenzbedingung in zwei Exportmodule, die jeweils einem separaten Datenfluss entsprechen (`APSY_SV` und `APSY_KV`). Wird im Dokumentationsmodul `APSY_PCI` im Feld `STATUSLAPSY` beispielsweise 2 oder 4 kodiert, wird dieses nach Abschluss der Dokumentation in das Exportmodul `APSY_KV` transformiert. Damit wird die korrekte Datenannahmestelle definiert, an die der Datensatz zu übermitteln ist – in diesem Fall die KV.

Das Attribut `type_QS_data` gibt Auskunft darüber, welchem Datentyp dies im XML-Schema entspricht.

Beispiel:

Beim Exportmodul `APSY_KV` ist der Datentyp im XML-Schema `qs_data_apsy_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.

2.5.2 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.

⁶⁴ Gemeint ist eine Follow-up-Erhebung mittels PID-Verfahren.

⁶⁵ Gemeint ist ein Follow-up-Verfahren anhand von PID-Daten.

Zusatzfelder⁶⁶

Ein Exportfelddatensatz beginnt immer mit den folgenden Zusatzfeldern:

RegistrierNr	=	Registriernummer des Dokumentationssystems (Länderkode + Registrierkode) ⁶⁷
Vorgangsnr	=	Vorgangsnummer
VorgangsnrGuid	=	Vorgangsnummer (GUID)
VersionNr	=	Versionsnummer
Storno	=	Stornierung eines Datensatzes (inkl. aller Teildatensätze)
Modul	=	Bezeichnung des Exportmoduls
Bogen	=	Teildatensatz (Bogen)
DokAbschlDat	=	Dokumentationsabschlusssdatum

Ein neuer Teildatensatz beginnt mit den Zusatzfeldern `RegistrierNr`, `Vorgangsnr`, `VorgangsnrGuid` und `VersionNr`.

Zusatzfelder, welche nicht in der Datenfelddbeschreibung (Tabelle `BogenFeld`) eines Moduls enthalten sind, werden von der QS-Dokumentationssoftware ausgefüllt.⁶⁸

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.
- 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.

**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.

⁶⁶ Dabei ist zu beachten, dass die Feldnamen beim Export durch die XML-Elemente zu ersetzen sind

⁶⁷ Für die Datenübermittlung an KVen und DAS-SV gelten andere Regelungen als bei der Datenübermittlung an die LKG.

⁶⁸ Hier gilt also nicht der Grundsatz, dass Felder nicht vorbelegt sein dürfen.

Eine vollständige Liste der möglichen Zusatzfelder findet sich in der Tabelle `ZusatzFeld` der Spezifikationsdatenbank zur QS-Dokumentation. Zusatzfelder sind in Tabelle 57 definiert.

Tabelle 57. Struktur der Tabelle `ZusatzFeld`

Feldname	Datentyp	Bemerkung
<code>idZusatzFeld</code>	INTEGER	Primärschlüssel
<code>name</code>	KURZER TEXT	Technischer Name (muss eindeutig sein)
<code>bezeichnung</code>	KURZER 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</code> = WAHR, 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.
<code>sortierNr</code>	INTEGER	Sortiernummer

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 2.7.1).

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 58. Struktur der Tabelle *Ersatzfeld*

Feldname	Datentyp	Bemerkung
idErsatzFeld	INTEGER	Primärschlüssel
name	KURZER TEXT	Technischer Name (muss eindeutig sein)
bezeichnung	KURZER TEXT	Bezeichnung des Ersatzfeldes
formel	KURZER TEXT	Berechnungsformel der Ersatzfelder
fkBasisTyp	INTEGER	Obligatorischer Fremdschlüssel zum Basistyp
fkSchluessel	INTEGER	Optionalen Fremdschlüssel zum Schlüssel

Beispiel:

Das Ersatzfeld „postoperative Verweildauer“ wird folgendermaßen berechnet:

```
poopvwdauer = ENTLDATUM - OPDATUM
```

Ersatzfelder, die nicht berechnet werden können

Es kann auch vorkommen, dass Ersatzfelder für einen Datensatz nicht berechnet werden können, weil die der Berechnung zugrundeliegenden 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`).

Anonymisierungsvorschriften

Die Anonymisierung von Datenfeldern wird wie aus der folgenden Tabelle 59 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 59. Struktur der Tabelle *ErsatzFuerFeld*

Feldname	Datentyp	Bemerkung
idErsatzFuerFeld	INTEGER	Primärschlüssel

Feldname	Datentyp	Bemerkung
fkFeld	INTEGER	Optionaler Fremdschlüssel zur Tabelle Feld
fkBogenFeld	KURZER TEXT	Optionaler Fremdschlüssel zur Tabelle BogenFeld
fkErsatzFeld	KURZER TEXT	Obligatorischer Fremdschlüssel zur Tabelle ErsatzFeld
parametrierbar	BOOL	Kennzeichen, ob das Feld parametrierbar ist

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.

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.

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 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 Listefeldern ist den Abfragen `Exportfelder` und `ExportfelderFürEinModul` zu entnehmen (`ExportfelderFürEinModul.elemente`).

- Zusatzfelder der Tabelle `ExportFormat` sind Muss-Felder, außer wenn sie Schlüsselfelder mit einem `Jeer`-Schlüssel (z. B. das Zusatzfeld `Storno`) sind.

**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.

2.6 Versionierung

Im Folgenden werden die Tabelle `Version`, der Abgleich zu vorherigen Versionen, die Abgrenzung zwischen Spezifikationsjahren und Datensatzformaten sowie die Version von Exportverfahren und -dateien beschrieben.

2.6.1 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 Spezifikationsjahr (z. B. Aufnahme zwischen dem 1. Januar 2019 und dem 31. Dezember 2019).

Die QS-Dokumentationssoftware eines Spezifikationsjahres 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.

**Hinweis**

Grundsätzlich baut eine QS-Dokumentationsdatenbank für ein Spezifikationsjahr auf der QS-Dokumentationsdatenbank aus dem vorherigen Spezifikationsjahr auf. Das Delta wird in den Tabellen `DeltaAttribut`, `DeltaGeloescht` und `DeltaNeu` dargestellt. Sollte

nach der ersten Veröffentlichung der QS-Dokumentationsdatenbank für ein Spezifikationsjahr noch ein Update der QS-Dokumentationsdatenbank aus dem vorherigen Spezifikationsjahr notwendig werden, dann divergieren beide Datenbanken. Würde man beide Datenbanken kombinieren, dann wären die IDs nicht garantiert rechtseindeutig. Eine `idVersion` könnte dann zum Beispiel auf unterschiedliche Werte in `QSDOK.Version` verweisen.

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`⁶⁹ verwaltet. Das Attribut `gueltig` zeigt die gültige Version der Datenbank an. Es darf nur eine einzige Version als gültig markiert sein.

Beispiel:

Die Version 2027 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 2027 V02 ist die Version 2027 V01. Da ab dem Spezifikationsjahr 2022 in der Tabelle `Version` die Vorjahresversionen nicht mehr enthalten sind, wird für die erste Version eines Spezifikationsjahres keine Vorgängerversion angezeigt.

2.6.2 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 (Spezifikation 2019):

Die Datenbank 2026_BASIS_QSF_V01 stellt die Änderungen im Vergleich zur letzten gültigen Spezifikationsdatenbank des Vorjahres 2025_BASIS_QSF_V08 dar.

⁶⁹ Der Status „Service Release zur finalen Spezifikation“ wird seit der Spezifikation 2014 nicht mehr verwendet.

Die Datenbank 2026_BASIS_QSF_V02 stellt zusätzlich die Änderungen im Vergleich zur Vorversion 2026_BASIS_QSF_V01 dar.

Es gibt drei Delta-Tabellen, die die neuen Entitäten, die gelöschten Entitäten und die geänderten Attributwerte weiterbestehender Entitäten aufzeigen.

Neue Entitäten

Über die Attribute `id` und `fkTabellenStruktur` in der Tabelle `DeltaNeu` (siehe Tabelle 60) ist die Zuordnung zu den Entitäten der Datenbank möglich.

Tabelle 60. Struktur der Tabelle *DeltaNeu*

Feldname	Datentyp	Bemerkung
<code>idDeltaNeu</code>	INTEGER	Primärschlüssel
<code>id</code>	KURZER TEXT	ID der Entität, die eingefügt wurde
<code>fkTabellenStruktur</code>	ZAHL	Bezug zur Tabelle, in die die Entität eingefügt wurde
<code>bemerkung</code>	LANGER TEXT	Begründung für die Ergänzung

Beispiel:

Die Zeile `idDeltaNeu = 47` enthält in der Spezifikation 2027 V01 folgende Werte:

`Id = 9640`

`fkTabellenStruktur = <BogenFeld>`

`bemerkung = APSY.B.GENEHMTS [3]`

Dies bedeutet, dass in der Tabelle `BogenFeld` ein neuer Eintrag mit `idBogenFeld = 9640` 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 61) ist die Zuordnung zu den Attributen der Entitäten der Datenbank möglich.

Tabelle 61: Struktur der Tabelle *DeltaAttribut*

Feldname	Datentyp	Bemerkung
<code>idDeltaAttribut</code>	AUTOWERT	Primärschlüssel
<code>id</code>	KURZER TEXT	ID der Entität, die geändert wurde

Feldname	Datentyp	Bemerkung
fkTabellenFeldStruktur	ZAHL	Bezug zum Attribut einer Tabelle, in der die Entität geändert wurde
alterInhalt	LANGER TEXT	Alter Inhalt der geänderten Entität in der letzten finalen Spezifikation
neuerInhalt	LANGER TEXT	Neuer Inhalt dieser Entität in der aktuellen Spezifikation
bemerkung	LANGER TEXT	Begründung für die Ergänzung

Beispiel:

Die Zeile `idDeltaAttribut = 153` enthält in der Spezifikation 2027 V01 folgende Werte:

`Id = 9293`

`Tabellenattribut = <BogenFeld.fkMussKann>`

`alterInhalt = K`

`neuerInhalt = M`

`bemerkung = APSY.B.BEGLTP [13]` Das heißt, dass in der Zeile `idBogenfeld = 9293` der Tabelle `BogenFeld` das Attribut `MussKann` von `K` auf `M` geändert wurde. Die Spalte `bemerkung` enthält ggf. weitere Hinweise zum Kontext der Änderung.

Gelöschte Entitäten

Über die Attribute `id` und `fkTabellenStruktur` in der Tabelle `DeltaGeloesch` (siehe Tabelle 62) ist die Zuordnung zu den Entitäten der Datenbank möglich.

Tabelle 62: Struktur der Tabelle *DeltaGeloesch*

Feldname	Datentyp	Bemerkung
<code>idDeltaGeloesch</code>	INTEGER	Primärschlüssel
<code>id</code>	KURZER 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>	LANGER TEXT	Ggf. Bemerkung zur Ergänzung

Beispiel:

Die Zeile `idDeltaGeloescht = 97` enthält in der Spezifikation 2027 V01 folgende Werte:

`Id = 920`

`fkTabellenStruktur = <Abschnitt>`

`bemerkung = APSY-'Therapieverlauf' [1:41 - 84]` Das heißt, dass aus der Tabelle `BogenFeld` der Vorgängerdatenbank das `BogenFeld` mit der `id = 920` 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 63: 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.

2.6.3 Abgrenzung zwischen Spezifikationsjahren 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.

■

**Achtung**

Dem Spezifikationsjahr 2027 zugeordnete Fälle müssen im Format der Spezifikation 2027 an die Datenannahmestellen gesendet werden.

Die Exportzeiträume sind den Attributen `Exportmodul.fkExportzeitraumEntlSJ` und `Exportmodul.fkExportzeitraumEntlSJ1` der Datenbank zur QS-Dokumentation zu entnehmen.

2.6.4 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.

2.7 Administrative Objekte

Die Datenbank für QS-Dokumentation beinhaltet neben den Dokumentationsobjekten eine Gruppe von Tabellen, die Hilfestellungen für die Einhaltung eines korrekten Datenflusses und standardisierter Prüfprozesse geben.

Zu den administrativen Objekten gehören Mapping-Informationen mit denen QS-Daten in Form von Metainformationen zu administrativen Zwecken außerhalb des eigentlichen QS-Containers (dem Element `<qs-data>`) im XML verortet werden können (Abschnitt B 2.7.1). Außerdem gibt es mit der Abfrage `vPruefung` eine Auflistung von Prüfungen, die für den jeweiligen Datenservice zu implementieren sind (Abschnitt B 2.7.3). 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.

Für die Leistungserbringer relevante Empfänger der Datenlieferungen sind in einer separaten Datenbank aufgelistet (Abschnitt B 2.7.2). Sie enthält auch Informationen über die zu verwendenden XML-Schlüssel.

2.7.1 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 hierfür relevanten Tabellen sind in der Abbildung 14 dargestellt.

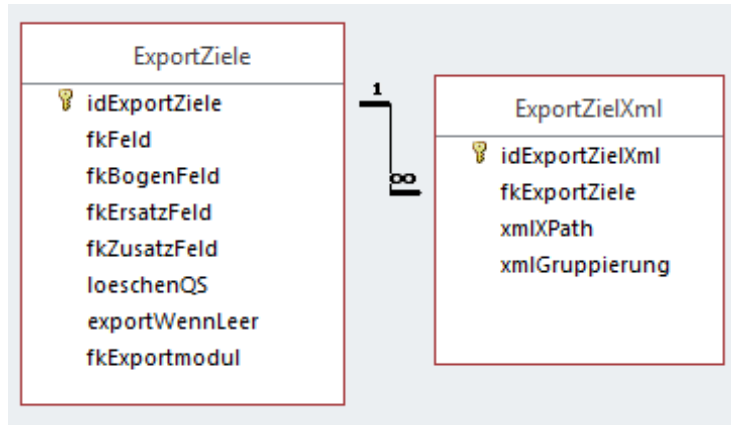


Abbildung 14: Beziehungen der administrativen Objekte – Mapping-Informationen

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.

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.

idExportZielXml	fkFeld	fkBogenFeld	fkErsatzFeld	fkZusatzFeld	loeschenQS	exportWennLeer	xmlXPath	xmlGruppierung	fkExportModul
2	IKNRKH			Modul	☒	☐	ancestor::data_container/care_provider/IKNR/@V	☒	
3				Modul	☐	☐	ancestor::cases/@module	☒	
4				Modul	☐	☐	case_admin/module/@V	☐	
5				Modul	☐	☐	qs_data/@module	☐	
7			versichertenidkv		☒	☐	patient/pid/VERSICHERTENIDNEU/@V	☐	
8			kasseknr2stellen		☐	☐	patient/@twodigitik	☐	
10	BSNRAMBU				☒	☐	ancestor::data_container/care_provider/BSNRAMBU	☒	
11	NBSNRAMB				☒	☐	ancestor::data_container/care_provider/NBSNRAMB	☒	
12	LANR				☒	☐	ancestor::data_container/care_provider/LANR/@V	☒	
13				RegistrierNr	☒	☐	ancestor::root/header/provider/@registration	☐	
14				VersionNr	☒	☐	case_admin/version/@V	☐	
15				Vorgangsnr	☒	☐	case_admin/id/@V	☐	
16				Storno	☒	☐	case_admin/action/@V="delete"	☐	
17			opquartal		☒	☐	case_admin/quarter/@V	☐	PCI_KV
18			vornameMutter		☒	☒	patient/perineo_pid/vorname_mutter/@V	☐	
19			nachnameMutter		☒	☒	patient/perineo_pid/nachname_mutter/@V	☐	
22			gebdatumNeo		☒	☒	patient/perineo_pid/GEBDATUMK/@V	☐	
23			gebdatumGeb		☒	☒	patient/perineo_pid/GEBDATUMK/@V	☐	
24			versichertenidkv		☒	☒	patient/perineo_pid/VERSICHERTENIDNEUK/@V	☐	NEO

Abbildung 15: Ergebnis der Abfrage `vExportZieleXml` (Auszug)

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
☒	☐		1 /root/body/data_container/care_provider/IKNR/@V	☒
☐	☐		2 ancestor::cases/@module	☒
☐	☐		2 case_admin/module/@V	☐
☐	☐		2 qs_data/@module	☐
☒	☐		4 patient/pid/VERSICHERTENIDNEU/@V	☐

Abbildung 16: Beispiel für XPath-Ausdrücke in der Tabelle `ExportZieleXml` in Verbindung mit weiteren Informationen

Einen ergänzenden Überblick über die zu exportierenden Felder eines Exportmoduls liefert die Tabelle `ExportFormatExportModul`. Sie zeigt alle möglichen Exportfelder für einen Datensatz (case) und informiert über deren Häufigkeit, Typ und xpath.

Tabelle 64: Struktur der Tabelle `ExportFormatExportModul`

Feldname	Bemerkung
<code>idExportFormatExportModul</code>	Primärschlüssel
<code>Exportmodul_name</code>	name aus Tabelle <code>Exportmodul</code>
<code>qs_data_xsi_type</code>	<code>type_QS_data</code> oder <code>type_QS_data_mds</code> aus Tabelle <code>Exportmodul</code> (Wenn Wert null, dann führt <code>xpath</code> zu keinem Kindelement von <code>qs_data</code> .)
<code>Bogen_name</code>	name aus Tabelle <code>Bogen</code> (Wenn Wert null, dann führt <code>xpath</code> zu keinem Kindelement von <code>qs_data</code> .)
<code>fkBogen</code>	Fremdschlüssel zur Tabelle <code>Bogen</code>
<code>BogenZahl_bzgl_mutterbogen</code>	Anzahl der Teildatensätze/Bögen bezogen auf den Mutterbogen (<code>QSDOK.Bogen.fkBogenZahl</code>)
<code>Bogen_mutterbogen_name</code>	name aus Tabelle <code>Bogen</code>
<code>fkMutterbogen</code>	Schlüssel zum Mutterbogen
<code>exportfeld</code>	name entweder aus Tabelle <code>Zusatzfeld</code> , <code>Ersatzfeld</code> oder <code>Feld</code>
<code>lfdNr</code>	Sortierhilfe innerhalb eines Bogens; untere Grenze kann größer als 1 sein (lfdNr 4 kann bspw. dem <code>xpath child::*[1]</code> entsprechen); wenn Wert gleich 0, dann führt <code>xpath</code> zu keinem Kindelement von <code>qs_data</code> .
<code>maxAnzahl</code>	maximale Anzahl des Exportfeldes auf dem Bogen (Wenn <code>fkBogen</code> gleich null, dann maximale Anzahl im Datensatz.)
<code>BasisTyp_name</code>	name aus Tabelle <code>BasisTyp</code>
<code>fkBasisTyp</code>	Fremdschlüssel zur Tabelle <code>BasisTyp</code>
<code>maxZeichenlaenge</code>	maximale Zeichenlänge
<code>xpath</code>	<code>xpath</code> innerhalb der Exportdatei (Ausgangspunkt bzw. current node ist <code><case></code>)
<code>fkMussKann</code>	Fremdschlüssel zur Tabelle <code>MussKann</code>
<code>fkBogenFeld</code>	Fremdschlüssel zur Tabelle <code>BogenFeld</code>
<code>BogenFeld_name</code>	name aus Tabelle <code>BogenFeld</code>

Feldname	Bemerkung
BogenFeld_bezeichnung	bezeichnung aus Tabelle BogenFeld
fkErsatzFeld	Fremdschlüssel zur Tabelle ErsatzFeld
ErsatzFeld_name	name aus Tabelle ErsatzFeld
ErsatzFeld_bezeichnung	bezeichnung aus Tabelle ErsatzFeld
fkZusatzFeld	Fremdschlüssel zur Tabelle ZusatzFeld
ZusatzFeld_name	name aus Tabelle ZusatzFeld
ZusatzFeld_bezeichnng	bezeichnung aus Tabelle ZusatzFeld
fkModul	Fremdschlüssel zur Tabelle Modul
Modul_name	name aus Tabelle Modul
fkExportZiele	Fremdschlüssel zur Tabelle ExportZiele
fkExportmodul	Fremdschlüssel zur Tabelle ExportModul
fkExportZielXml	Fremdschlüssel zur Tabelle ExportZielXml
Bogen_sortierNr	sortierNr aus Tabelle Bogen
fkExportformat	Fremdschlüssel zur Tabelle ExportFormat

Hinweis

XML-Kindelemente von `qs_data`-Elemente werden in der Exportdatei immer gesetzt, sobald der Teildatensatz in der Exportdatei vorhanden ist. Sollte bspw. kein Wert für Exportfeld `feld_2` vorliegen, dann ist die Abbildung in der Exportdatei:

```
<feld_1V="foo"/>
```

```
<feld_2 />
```

```
<feld_3 />
```

Falsch wäre und schemaverletzend:

```
<feld_1V="foo"/>
```

```
<feld_3 />
```

Die Optionalität wird also durch das XML-Attribut V abgebildet.

Bei `qs_data` Kindern mit `maxAnzahl > 0` muss nur das erste XML-Kindelement geschrieben werden, wenn keine Werte zu exportieren sind.

2.7.2 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 hier zu finden:

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

Die einzelnen Datenservices werden zusammen mit den E-Mail-Adressen⁷⁰, an die die Exportdateien zu versenden sind, in der Tabelle *Datenservice* abgebildet:

fkInstitution	funktion	sektor	datenabsender	email	kv_connect	upload
QiG-BW	DAS	Krankenhaus	LE	daten@qigbw.de	☐	☐
BAQ	DAS	Krankenhaus	LE	daten@baq-bayern.de	☐	☐
PGS-Sachsen	DAS	Krankenhaus	LE	daten@qs-sachsen.de	☐	☐
KGSH	DAS	Krankenhaus	LE	qsdaten@kgsh.de	☐	☐
BQS-Institut	DAS	Krankenhaus	LE	xmldaten@bqs-institut.de	☐	☐
IQTIG	BAS		VST		☐	☐
VST-PSN	VST		DAS		☐	☐
IQTIG	DAS	Krankenhaus	LE	daten@iqtig.org	☐	☐
VST-DAS	DAS	selektiv	LE		☐	☐
KV-Bayern	DAS	kollektiv	LE		☒	☒
KV-Baden-Württemberg	DAS	kollektiv	LE		☒	☒

Abbildung 17: Tabelle *Datenservice*

Die Zuordnung eines Datenservices zu einer Region abhängig vom Modul sowie der zu verwendende Schlüssel sind ab der Spezifikation 2022 in der Tabelle *DatenserviceRegionExportModul* festgelegt:

fkRegion	fkDatenservice	fkExportModul	fkSchluessel
BE / Berlin	DAS / KV-Berlin	DIAL_KV	Pub_key_Datenannahmestelle_BE_KV.pub
BE / Berlin	DAS / KV-Berlin	PCI_KV	Pub_key_Datenannahmestelle_BE_KV.pub
BE / Berlin	DAS / QB-Berlin	LLSFU_DeQS	Pub_key_Datenannahmestelle_BE_LKG.pub
BE / Berlin	DAS / QB-Berlin	DIAL_LKG	Pub_key_Datenannahmestelle_BE_LKG.pub
BE / Berlin	DAS / QB-Berlin	CHE	Pub_key_Datenannahmestelle_BE_LKG.pub
BE / Berlin	DAS / QB-Berlin	DEK	Pub_key_Datenannahmestelle_BE_LKG.pub
BE / Berlin	DAS / QB-Berlin	PCI_LKG	Pub_key_Datenannahmestelle_BE_LKG.pub
BE / Berlin	DAS / QB-Berlin	15/1	Pub_key_Datenannahmestelle_BE_LKG.pub
BE / Berlin	DAS / QB-Berlin	10/2	Pub_key_Datenannahmestelle_BE_LKG.pub

Abbildung 18: Verfahrensbezogene Datenservices

Eine zusammenfassende Darstellung der wichtigsten Merkmale ergeben sich aus den beiden Abfragen *DatenserviceExportverfahren* und *DatenserviceExportModule*. Die erste Abfrage aggregiert auf Ebene der Exportverfahren, während die *DatenserviceExportModule* eine weitergehende Differenzierung für die einzelnen Exportmodule anbietet.

2.7.3 Prüfschritte

In der Gruppe administrative Objekte der QSDOK-Datenbank befindet sich die Abfrage *vPruefung*. *vPruefung* zeigt für die Institutionsarten jeweils die Prüfungen, die relevant für die Beurteilung der Güte der angelieferten Daten sind.

Tabelle 65: Felder der Abfrage *vPruefung*

Feldname	Bedeutung	Bezugselement im XML
Zielgruppe	Institutionsart, die die Prüfung ausführt	–

⁷⁰ Die KVen und die DAS-SV erhalten die Daten in der Regel auf anderen Übertragungswegen.

Feldname	Bedeutung	Bezugselement im XML
idPruefung	Referenz auf die konkrete einzelne Prüfung	–
pruefung	Kurzbeschreibung der Prüfung	optional unter: validation_provider/validation_item/@description
dpp	True/False, Prüfung ist im Datenprüfprogramm enthalten (B 5.1)	–
meta_DS	meta_DS soll ausdrücken, ob das Prüfergebnis das Persistieren der Metadaten (case/case_admin) eines Datensatzes verhindert. Wenn eine verletzte Prüfung die Eigenschaft meta_DS = true hat, dann werden die Metadaten nicht persistiert. Sind nur Prüfungen mit meta_DS = false verletzt, müssen die Metadaten aus case/case_admin unabhängig von status_document und status_case persistiert werden.	
beschreibung	ggf. erweiterte Erläuterungen zur Prüfung	–
fkFehlermeldung	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 In der Exportdatei ist ERROR auf H abzubilden und WARNING auf W.
regelverletzung	Bedingungen, die zu einer Regelverletzung führen	–
fehlermeldung	Standardisierte Fehlermeldung	validation_item/error/error_message/@V
parameter	Parameter für Fehlermeldung	–

Feldname	Bedeutung	Bezugselement im XML
protokoll-Level	Zuordnung der Prüfung: DK-> betrifft Dokument DS-> betrifft Datensatz	-
datenfluss	pid, sv, vpb	
verursacher	Verursacher des Fehlers	error/@originator
sortierung	mögliche Prüfungsreihenfolge	

Das Attribut Prüfung . dpp

Das Attribut `dpp` ist `TRUE`, wenn die entsprechende Fehlerprüfung über das Datenprüfprogramm abgebildet wird. Hierbei muss die Prüfung (mit der jeweiligen Prüfnummer `idFehlermeldung`) nicht eins-zu-eins im Datenprüfprogramm referenziert sein. Das Attribut ist auch dann `TRUE`, wenn es im Datenprüfprogramm eine analoge Prüfung gibt, die denselben Sachverhalt prüft und sicherstellt, dass keine Regelverletzung auftritt. Hierbei kann es sich auch um eine Schemaprüfung handeln. Das Attribut `dpp` dient dem Zweck (z.B. einer Datenannahmestelle) eine Übersicht über die Prüfungen zu geben, die bereits über das Datenprüfprogramm geprüft werden und keine separate (z.B. durch die Datenannahmestelle implementierte) Zusatzprüfung erfordern.

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.

Hinweis:

Dieses Kapitel ist nicht relevant für das QS-Verfahren Ambulante Psychotherapie.

4 XML-Schema

Die XML-Schema-Datei (XSD) ist eine Empfehlung des W3C⁷¹ 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.

⁷¹ <http://www.w3.org/XML/Schema>

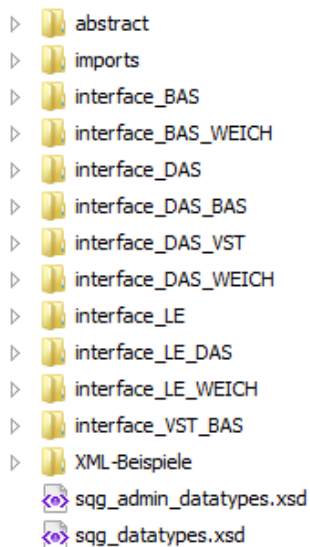


Abbildung 19: Dateiordner der Schnittstellen-Schemata

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

Beispiel:

Die leistungserbringeridentifizierenden Daten existieren gemäß Datenflussmodell der Richtlinien des G-BA 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

Die in der Spezifikationskomponente XML-Schema enthaltene `readme.csv` listet die Schema-Dateien auf, 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.

Um nach einer Schemavalidierung der XML-Dateien die Weiterverarbeitung und dementsprechend die spezifikationskonforme Protokollierung auf Datensatzebene weiterhin zu ermöglichen,

wurden zusätzlich „weiche“ Schemata für die Schnittstellen LE und DAS eingeführt. Diese weichen Varianten werden ausschließlich mit dem Datenprüfprogramm verwendet (Abschnitt B 5.1). Es ist dabei zu beachten, dass nur die DAS, die die QS-Daten entschlüsseln, diese weiche Variante benötigen. Diese sind LKG, KV und BAS.

In der folgenden Tabelle werden Schema-Dateien aufgeführt, die im Rahmen der Rückprotokollierung Verwendung finden.

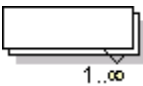
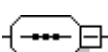
Tabelle 66: XML-Schemata für die Rückprotokollierung

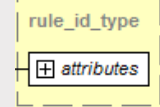
Schnittstelle	Schema	Verwendungszweck in Zusammenhang mit dem DPP
LE/DAS interface_LE_DAS	response_DAS_LE.xsd	Datenflussprotokoll der DAS
	response_receipt.xsd	Empfangsbestätigung der DAS
DAS/VST interface_DAS_VST	response_VST_DAS.xsd	Datenflussprotokoll der VST
	response_receipt.xsd	Empfangsbestätigung 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 67: 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

Symbol	Beschreibung
	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
	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 zugrundeliegenden 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/xmenc#"`

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/xmenc#"
>
```

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>`.

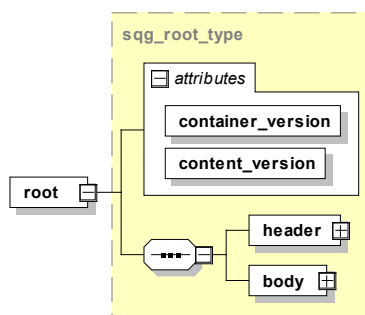


Abbildung 20: Root-Element und Kind-Elemente header und body

Das Root-Element hat zusätzlich zwei Attribute (Tabelle 68).

Tabelle 68: 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 beibehalten, um die Aufwärtskompatibilität zu gewährleisten. Ein XML-Dokument, das einen alten Wert dieses Attributs enthält, muss von der Datenannahmestelle zurückgewiesen werden.
content_version	content _version _Datentyp	required	1.0	Ist ein fixer Wert und definiert die aktuell gültige Versionsnummer des Inhalts der QS-Daten. Die Versionsnummer wird erhöht, wenn unterjährig das Schema unabhängig von der zugrundeliegenden Spezifikationsdatenbank geändert wird.

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.

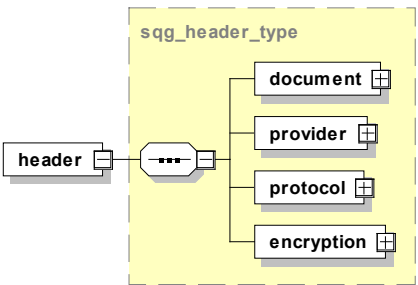


Abbildung 21: Aufbau des Elements header

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.

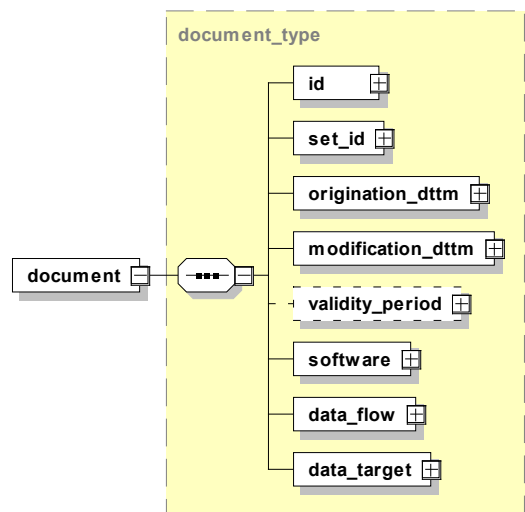


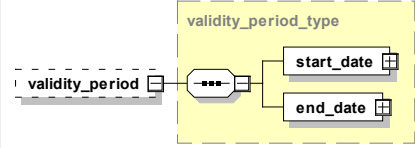
Abbildung 22: Aufbau des Elements document

Dieses Element hat weitere Kind-Elemente, die in der folgenden Tabelle beschrieben werden.

Tabelle 69: Kind-Elemente des Elements document

Kind-Elemente	Beschreibung
<id>	Nach Erstellung nicht mehr modifizierbar. Eindeutige ID des Dokuments, wird vom Dokumentenersteller als GUID erzeugt. ⁷²
<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

⁷² Ein Globally Unique Identifier oder kurz GUID ist eine global eindeutige Zahl mit 128 Bit (16 Bytes), die ein Dokument weltweit eindeutig identifiziert.

<code><validity_period></code>	 oder ein Quartal oder ein frei wählbarer Bereich sein. Zur Qualifizierung sind die Felder <code><start_date></code> und <code><end_date></code> entsprechend zu füllen.	Mithilfe des optionalen <code><validity_period></code> -Elements wird der Bezugszeitraum der Datei angegeben. Dies kann ein Jahr
--------------------------------------	---	--

`<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). Fehlt die Zeitzonenangabe, wird die lokale Zeitzone angenommen.

Beispiele (jeweils dieselbe Uhrzeit):

- 2023-06-30T21:32:52 (Angabe Zeitzone fehlt, daher Annahme lokale Zeitzone)
- 2023-06-30T21:32:52+02:00
- 2023-06-30T19:32:52Z

Element header/document/software

Sammelement für Angaben zur eingesetzten QS-Software.

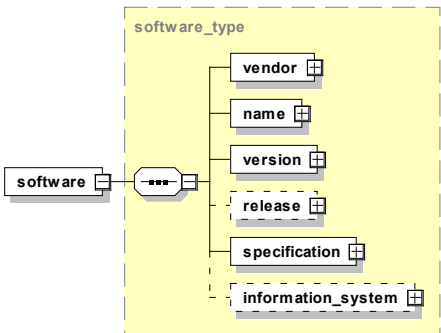


Abbildung 23: Aufbau des Elements software

Dieses Element enthält Kind-Elemente, die in der folgenden Tabelle beschrieben werden.

Tabelle 70: 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 Spezifikationsdatenbank (QSDOK)
<information_system>	Enthält Angaben zum eingesetzten Informationssystem (PVS).

Element header/document/software/information_system

Sammelement für Angaben zum eingesetzten Informationssystem (PVS).

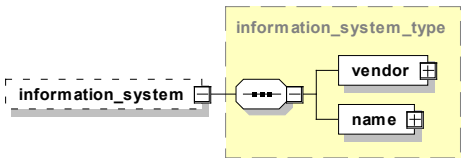


Abbildung 24: Aufbau des Elements information_system

Tabelle 71: Kind-Elemente des Elements information_system

Kind-Elemente	Beschreibung
<vendor>	Enthält Informationen über den Softwarehersteller.
<name>	Enthält den Softwarenamen der eingesetzten Software.

Das Attribut //software/vendor/@registration ist veraltet und optional. Eine Registrierung der Software ist nicht mehr notwendig bzw. möglich.⁷³

Element header/document/data_flow

Dieses Element gibt an, für welchen Datenfluss (Datenannahmestelle) dieses Dokument erzeugt wurde.

⁷³ Die Registriernummer für eine Software ist nicht die Registriernummer des Dokumentationssystems / der Softwareinstanz (//header/document/software/vendor/@registration != //header/provider/@registration). Das Datenfeld „RegistrierNr“ (siehe QSDOK.ZusatzFeld) entspricht dem Attribut //header/provider/@registration (siehe QSDOK.ExportZielXml).

Tabelle 72: 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 jeweils zuständige Datenannahmestelle des Landes 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
Stellungnahmeverfahren	<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 73: 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.

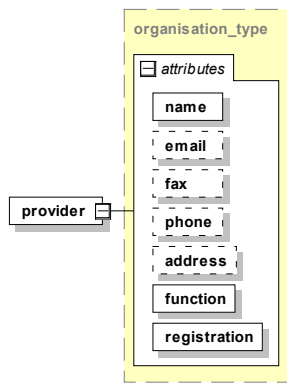


Abbildung 25: Aufbau des Elements provider

Tabelle 74: 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, Leistungserbringer, Soft- warehersteller, Versendestelle, Ver- trauensstelle, undefined
registration	registration_type	required	Registrierungsnummer. Sollte für die Datenübertragung an die DAS (DAS-SV) keine Registrie- rungsnummer erforderlich, ist das Element mit einem Dummy-Wert „000“ zu füllen. Sollte für die Datenübertragung an die DAS (KV) keine Registrierungs- nummer erforderlich sein, kann die- ses Element für den Datenfluss vom ambulanten Leistungserbringer an die KV weggelassen werden. Für alle folgenden Datenflüsse ist die Regist- riernummer 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.

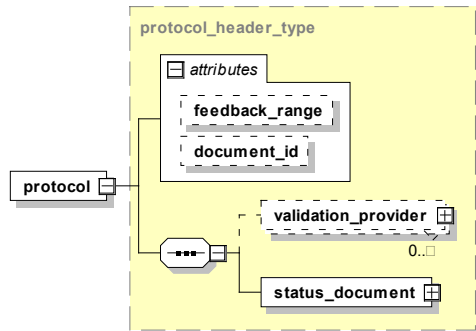


Abbildung 26: Aufbau des Elements header/protocol

Dieses Element hat zusätzlich zu den optionalen Attributen feedback_range und document_id zwei Kind-Elemente:

<validation_provider> und <status_document>

Tabelle 75: 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 document_id 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.

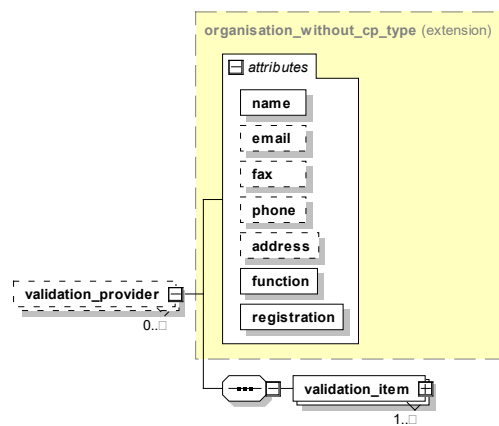


Abbildung 27: Aufbau und Kind-Elemente des Elements validation_provider

Tabelle 76: 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.

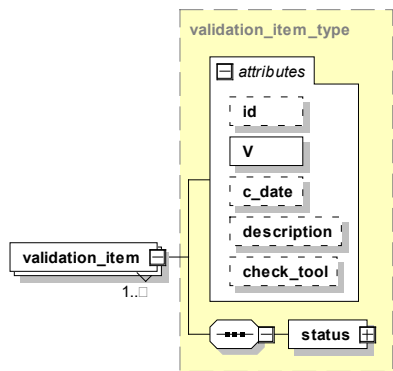


Abbildung 28: Aufbau und Kind-Elemente des Elements `validation_item`

Tabelle 77: 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 Datensebene (Element <code><case></code>), 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 <code>sqq_protocol.xsd</code> 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 <code>vPruefung</code> 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 gleichbleibt. `ERROR` bedeutet, dass das Dokument komplett zurückgewiesen werden muss.

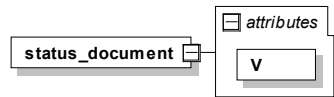


Abbildung 29: Aufbau des Elements status_document

Tabelle 78: 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.

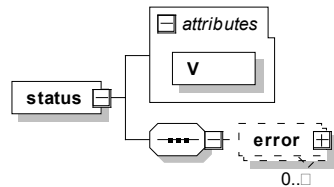


Abbildung 30: Aufbau und Kind-Elemente des Elements status

Tabelle 79: 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:

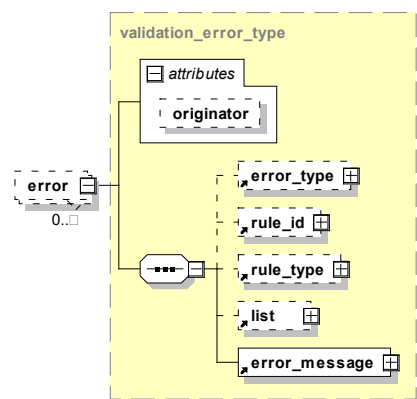


Abbildung 31: Aufbau des Elements error

Tabelle 80: Attribut des Elements error

Name	Type	Use	Beschreibung
originator	enum_organisation_type	optional	Mögliche Werte: Bundesauswertestelle, Datenannahmestelle, Leistungserbringer, Softwarehersteller, Versendestelle, Vertrauensstelle, undefined

Tabelle 81: 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 (id-Fehlermeldung)
<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 ⁷⁴ : EXPORT = Formatfehler der Exportdatei DOPPELT = bereits vorhandener Datensatz wird erneut übermittelt TDS = Vollständigkeit und Version der Teildatensätze WERT = Wertebereichsverletzung REGEL = Plausibilitätsverletzung

⁷⁴ Die Ausprägung STEUER (= Formatfehler der Steuerdatei) wurde im Rahmen eines csv-Exportet eingeführt und wird seit Überführung in das XML-Format nicht mehr verwendet

Kind-Elemente	Beschreibung
	KOLLISION = Patientenpseudonym mit unterschiedlichen Alters-/Geschlechtsangaben LE = LE nicht entschlüsselbar FEHLT = Angabe fehlt PID = PID nicht entschlüsselbar (in Kombination mit Dechiffrierung von „validation_item“) QS = QS-Daten nicht entschlüsselbar (in Kombination mit Dechiffrierung von „validation_item“) IST_Statistik = Fehlende/falsche Angaben zur IST-Statistik SOLL_Statistik = Fehlende/falsche Angaben zur SOLL-Statistik

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 (XPack).

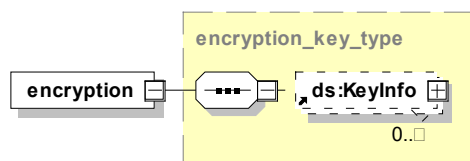


Abbildung 32: Aufbau und Attribute des Elements `encryption`(Krankenhaus)

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. 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 5.2) zu entnehmen.

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.

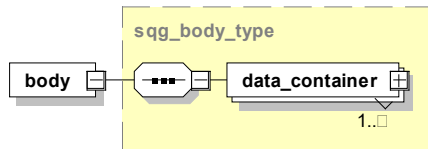


Abbildung 33: Aufbau des Elements `body`

Kind-Element `body/data_container`

Ein `<data_container>` ist einem bestimmten Leistungserbringer zuzuordnen. Da grundsätzlich mehr als ein `<data_container>` erlaubt ist, können ggf. auch mehrere `<data_container>` für mehrere Leistungserbringer verwendet werden, wenn das Dokument 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. Insbesondere durch die Pseudonymisierung von Standortangaben im KH-Bereich können auch von einem Krankenhaus mit mehreren Standorten mehrere `<data_container>` für Daten desselben Moduls sinnvoll sein. Wenn beispielsweise zwei leistungserbringende Standorte (L1, L2) und ein entlassender Standort (E1) in mehreren Fällen desselben Moduls dokumentiert wurden, dann können in einem Export zwei verschiedene `<data_container>` mit folgenden Standort-Kombinationen innerhalb des `<care_provider>`-Elementes angegeben werden:

- E1 + L1
- E1 + L2

Es ist also zu beachten, dass sämtliche Fälle (`<case>`-Elemente) stets der übergeordneten Kombination von Standortangaben zugeordnet werden.

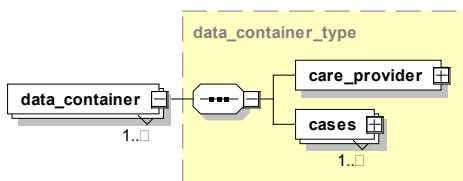


Abbildung 34: Aufbau des Elements `body/data_container`

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 (vertragsärztliche Praxis) durch einen unterschiedlichen Aufbau. Die Inhalte des `<care_provider>`-Elements ergeben sich aus Anwendung der Abfrage `vExportZieleXml` (Abschnitt 2.7.1) der QSDOK-Datenbank.

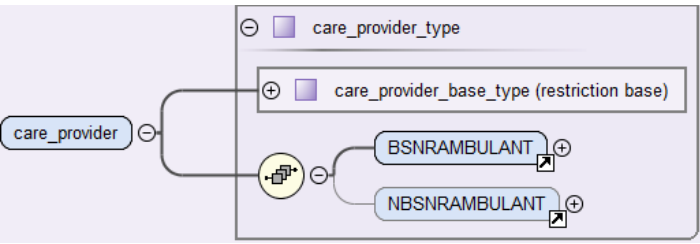


Abbildung 35: Aufbau des Elements `care_provider` – kollektiv-, selektivvertraglich

Die Kind-Elemente für die vertragsärztliche Praxis werden in der folgenden Tabelle beschrieben.

Tabelle 82: Leistungserbringeridentifizierende Daten im kollektiven, selektiven und stationären Bereich

Kind-Elemente	Beschreibung
Leistungserbringeridentifizierende Daten im kollektiven oder selektiven Bereich	
<code><BSNRAMBULANT></code>	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.
<code><NBSNRAMBULANT></code>	Nebenbetriebsstättennummer – NBSNR
<code><IKNR>⁷⁵</code>	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.
Leistungserbringeridentifizierende Daten im Bereich Krankenhaus	
<code><standortID></code>	entlassender Standort (6-stellige Standort_ID gemäß 3.2.3 Ebene 1: Krankenhaus-Standort, Anlage 1 zur Vereinbarung gemäß § 293 Abs. 6 SGB V (77 + StandortID + 0 + Einrichtungstyp) Ebene 1 Krankenhaus-Standort (6-stellig): Eindeutiges Kennzeichen für den Standort (Stellen 1-2: "77", Stellen 3-6 fortlaufende Vergabe durch Verzeichnisstelle.)
<code><standortIDaufn></code>	aufnehmender Standort (6-stellige Standort_ID)
<code><standortIDicd></code>	diagnostizierender Standort (6-stellige Standort_ID)
<code><standortIDops></code>	behandelnder Standort (6-stellige Standort_ID)

⁷⁵ Die IKNRKH ist ein optionales Element im kollektivvertraglichen Bereich. Derzeit ist dieses Element im KV-Bereich nicht zu verwenden.

Element body/data_container/cases

Container-Element für eine Liste von Fällen (Vorgängen) eines Exportmoduls. Für unterschiedliche Exportmodule müssen jeweils mehrere <cases> angelegt werden.

Die Ausweisung eines <cases>-Elements für Daten eines bestimmten Exportmoduls erfolgt über das Attribut module. Es ist zu beachten, dass das Modul MDS kein Exportmodul ist (siehe Abbildung 70).

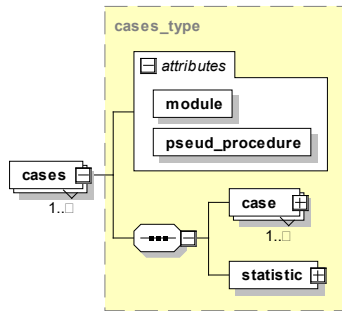


Abbildung 36: Aufbau des Elements cases

Tabelle 83: Attribute des Elements cases

Name	Type	Use	Beschreibung
module	enum_module_type	required	Exportmodul
pseud_procedure	enum_procedure_type	required	Zuordnung des Moduls zu einem Pseudonymisierungsverfahren (siehe ExportModul.fkPseudonymVerfahren [name]). Ist das Modul keinem Pseudonymisierungsverfahren zugeordnet, so ist das Attribut auf „undefined“ zu setzen.

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>".

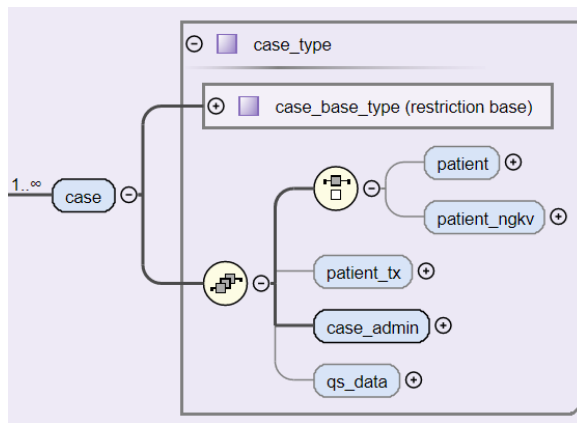


Abbildung 37: Aufbau des Elements case

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.

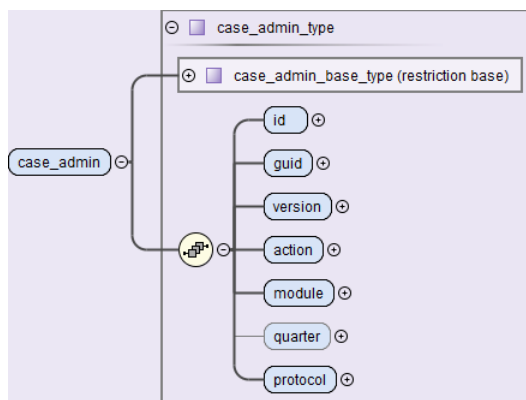


Abbildung 38: Aufbau des Elements case_admin

Im Folgenden werden die einzelnen Kind-Elemente beschrieben.

Tabelle 84: Kind-Elemente des Elements case_admin

Kind-Elemente	Beschreibung
<id>	Vorgangsnummer - diese Nummer kennzeichnet jeden dokumentierten Vorgang eines Dokumentationssystems eindeutig und zwar unabhängig vom angewandten Modul. Im einfachsten Fall könnte also die Vorgangsnummer um 1 erhöht werden, wenn ein neuer Vorgang angelegt wird. Im stationären Bereich, z. B. wenn während eines Krankenhausaufenthalts zwei QS-Dokumentationen eines Falles angelegt werden, müssen auch unterschiedliche Vorgangsnummern vergeben werden. Es ist falsch 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. Die Vorgangsnummer darf für die Datenannahmestelle nicht auf Personen zu beziehen sein. 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.
<guid>	36-stelliger pseudozufälliger Globally Unique Identifier (GUID) zur eindeutigen bundesweiten Identifizierung eines Vorganges. Die GUID lässt keine Rückschlüsse auf den Leistungserbringer oder den Patienten zu.
<version>	Enthält die Versionsnummer des Vorganges. Sie gibt an, die wievielte Version des Vorganges 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 Vorgang erneut eingesandt werden, so muss die Versionsnummer vom dokumentierenden System um 1 erhöht werden. Die neue Version des Vorganges wird bei der Entgegennahme geprüft und überschreibt bei Korrektheit die alte Version des Vorganges. Wenn die Datenannahmestelle den Vorgang mit derselben Versionsnummer ein zweites Mal erhält, so wird dieser zurückgewiesen.
<module>	Verfahrensabkürzung. Hier ist zu beachten, dass dieser Wert identisch zu den Attributwerten im Element <cases> und <qs_data> sein muss. Wenn die Datenannahmestelle unterschiedliche Modulbezeichnungen innerhalb eines <cases>-Elements erhält, wird die ganze Datenlieferung zurückgewiesen.
<quarter>	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.
<protocol>	Protokoll auf Vorgangsebene
<action>	Definiert die gewünschte Aktion, kann „create“, „update“ und „delete“ sein. „create“ ist beim ersten Export des Vorganges zu verwenden, weitere Exporte des Vorganges werden mit „update“ geliefert.

	Aus verschiedenen Gründen ist es im Datenfluss möglich, dass ein <code>update</code> eher die Datenannahmestelle erreicht als ein <code>create</code>. Daher sind <code>create</code> und <code>update</code> gleichzubehandeln. Entscheidend ist <code>case_admin/version/@V</code>.⁷⁶ Um den Vorgang zu stornieren, muss <code><action></code> auf „delete“ gesetzt werden. Die Datenannahmestelle wird dadurch veranlasst, den betreffenden Vorgang einschließlich aller Vorversionen und Teildatensätze als „storniert“ zu kennzeichnen. Der Stornovorgang wird in der Datenbestätigung protokolliert. Der Datensatz mit <code>delete</code> in <code>./action/@V</code> 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. Ein Stornoversuch eines noch nicht übermittelten Vorganges wird ebenfalls zurückgewiesen. Zur Stornierung eines Vorganges ist der Export der entsprechenden administrativen Daten <code><case>/<case_admin></code> zwingend erforderlich. QS-Daten <code><case>/<qs_data></code>, die ET-Nummer für das TX-Register <code><case>/<patient_tx></code>, PID <code><case>/<patient></code> oder <code><case>/<patient_ngkv></code> sind nicht erneut zu übermitteln.
--	--

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 2.7.1) in der QSDOK-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.

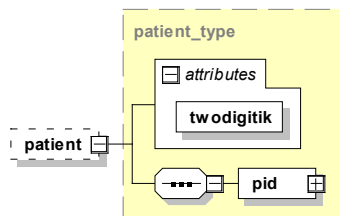


Abbildung 39: Aufbau des Elements patient

Sollten die verpflichtenden Attribute der `<patient>`-Kindelemente aufgrund von `QSDOK.ExportZiele.exportWennLeer` nicht existieren, dann ist das `<patient>`-Element vollständig wegzulassen. Bei einer Stornierung (`case_admin/action/@V = 'delete'`) ist ebenfalls das gesamte `<patient>`-Element wegzulassen. Dieselben Regeln gelten auch für die XML-Elemente `<patient_ngkv>` und `<patient_tx>`.

⁷⁶ Sollte ein `update` mit Version 2 bereits in der Datenannahmestelle persistiert sein, dann ist ein später eintreffendes `create` mit `case_admin/version/@V` gleich 1 abzulehnen (siehe auch QSDOK.vPruefung mit `idPruefung` 37 und 48). Käme ein weiteres `create` mit Version 3, würde keine Ablehnung erfolgen. Ebenso würde danach ein weiteres `create` mit Version 4 zu keiner Ablehnung führen.

Element patient_type/pid

Das Element `<pid>` dient dazu, die tatsächlichen PID aufzunehmen.

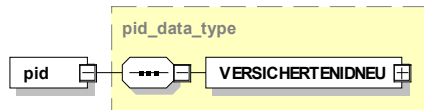


Abbildung 40: Aufbau des Elements `pid`

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 und `<GEBDATUMK>` für das Geburtsdatum des Kindes.

Element body/data_container/cases/patient_ngkv

Für die Module der Transplantationsmedizin werden PID-Daten auch für nicht gesetzlich versicherte Patienten erhoben, wenn eine Einwilligung dafür vorliegt. Wenn das Bogenfeld "Liegt eine wirksame Einwilligung des Patienten zur weiteren Übermittlung personenbezogener QSDaten (einschließlich ETNummer) an die Bundesauswertungsstelle vor?" dem Wert "1" (ja) entspricht, dann wird das Element `<patient_ngkv>` exportiert. Liegt keine Einwilligung vor, dann darf das Element `<patient_ngkv>` nicht im xml-Datensatz vorhanden sein.

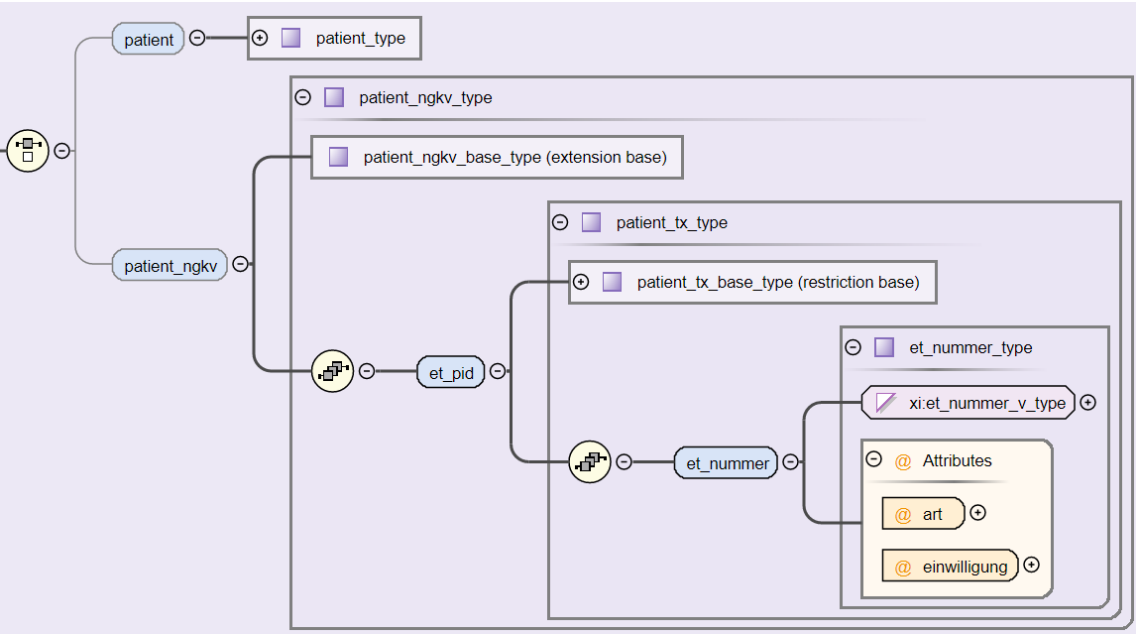


Abbildung 41: Aufbau des Elements patient_ngkv

Die nachfolgende Tabelle erklärt das Element `patient_ngkv/et_pid/et_nummer` und dessen Attribute `@art` sowie `@einwilligung`.

Tabelle 85: Kind-Elemente des Elements patient_ngkv/et_pid

Kind-Elemente	Beschreibung
<et-nummer>	Eurotransplant-Nummer des Typs String bestehend aus 6 Zahlzeichen. Bitte beachten Sie, dass die ET-Nummer im Inhalt des Elementes <et-nummer> steht und nicht in einem Attribut @V.
./@art	Das Attribut @art bestimmt, ob es die ET-Nummer des Spenders oder Empfängers ist.
./@einwilligung	Wenn das Bogenfeld "Liegt eine wirksame Einwilligung des Patienten zur weiteren Übermittlung personenbezogener QSDaten (einschließlich ETNummer) an die Bundesauswertungsstelle vor?" dem Wert "j" entspricht, dann ist der Wert das Attributes @einwilligung gleich "V" (für "vorhanden").

Das Element `<patient_ngkv>` ist mit dem öffentlichen Schlüssel der Vertrauensstelle zu verschlüsseln. Die Vertrauensstelle pseudonymisiert die ET-Nummer und leitet die pseudonymisierte ET-Nummer an die Bundesauswertungsstelle weiter.

Element body/data_container/cases/patient_tx

Für die Module der Transplantationsmedizin gilt, wenn das Ersatzfeld „Liegt eine wirksame Einwilligung des Patienten zur weiteren Übermittlung der Daten an das TX-Register vor?“ dem Wert

„V“ entspricht, dann wird das Element <patient_tx> exportiert. Ist der Wert des eben genannten Ersatzfeldes LEER, dann darf das Element <patient_tx> nicht im xml-Datensatz vorhanden sein.⁷⁷

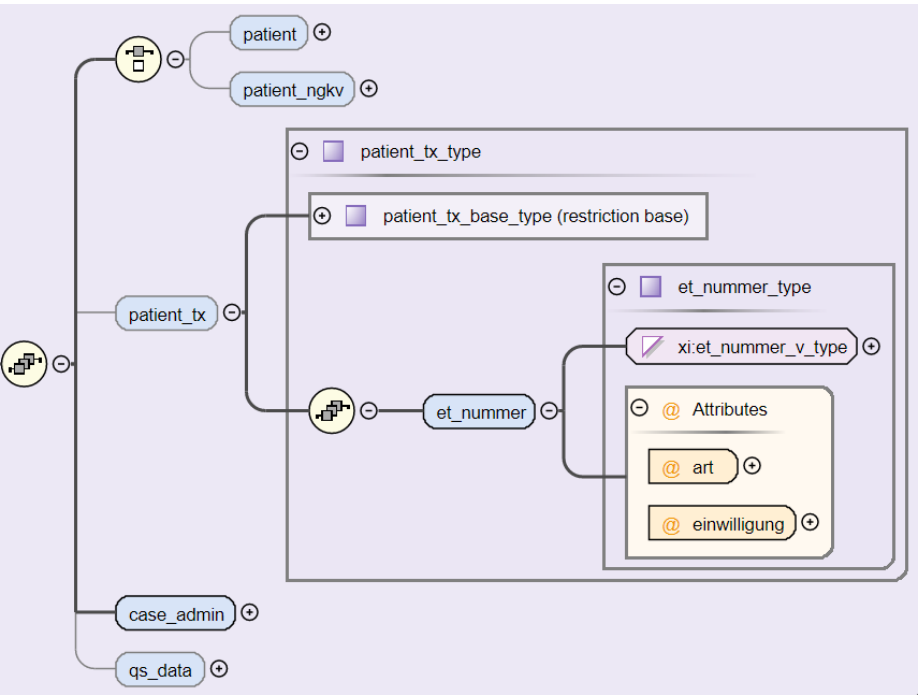


Abbildung 42: Aufbau des Elements patient_tx

Die nachfolgende Tabelle erklärt das Kindelement <et_nummer> und dessen Attribute @art sowie @einwilligung.

Tabelle 86: Kind-Elemente des Elements patient_tx

Kind-Elemente	Beschreibung
<et-num-mer>	Eurotransplant-Nummer des Typs String bestehend aus 6 Zahlzeichen. Bitte beachten Sie, dass die ET-Nummer im Inhalt des Elementes <et-nummer> steht und nicht in einem Attribut @V.
./@art	Das Attribut @art bestimmt, ob es die ET-Nummer des Spenders oder Empfängers ist.
./@einwil-ligung	Wert des aus der Syntaxfunktion einwMappingTX berechneten Ersatzfeldes „Liegt eine wirksame Einwilligung des Patienten zur weiteren Übermittlung der Daten an das TX-Register vor?“ (wenn ungleich LEER)

⁷⁷ Die Exportfelder etTXE/etTXS, etArtTXE/etArtTXS und etEinwTX sind als Muss-Felder definiert. Liegt keine Einwilligung des Patienten vor, wird gemäß definierten Syntaxfunktionen allen drei Exportfeldern ein Leerstring zugewiesen. In diesem Fall entfällt das Element <patient_tx> im xml-Datensatz. Sollten Softwareanbieter beim Erzeugen der Exportdatei einen Zwischenschritt implementiert haben, in dem vor Erzeugen einer spezifikationskonformen XML-Datei noch die drei Exportfelder unter Angabe von Leerstrings vorhanden sind, ist durch den Softwareanbieter sicherzustellen, dass die als Muss-Felder definierten Exportfelder nicht den Export verhindern.

Das Element `<patient_tx>` ist mit dem öffentlichen Schlüssel der Vertrauensstelle für die TX-Register zu verschlüsseln. Die Vertrauensstelle pseudonymisiert die ET-Nummer und leitet die pseudonymisierte ET-Nummer an das TX-Register weiter.

Element `case_Datentyp/case_admin/protocol`

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

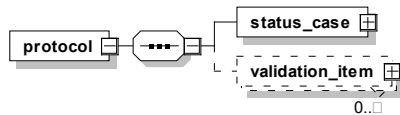


Abbildung 43: Aufbau des Elements `case_admin/protocol`

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 Bundesauswertungsstelle aufzunehmen.

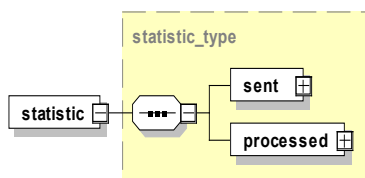


Abbildung 44: Aufbau des Elements `statistic`

Es ist nach der Prüfung bei der Bundesauswertungsstelle ein Teil des Rückprotokolls und besteht aus ähnlichen Kind-Elementen.

Tabelle 87: Kind-Element des Elements statistic

Kind-Element	Beschreibung
<sent>	Statistik über die von dem Datenlieferanten exportierten Datensätze. Es muss daher vom Datenlieferanten selbst ausgefüllt werden.
<processed>	Hat dieselbe Struktur wie <sent> und enthält das Ergebnis der Verarbeitung durch die Bundesauswertungsstelle.

Element statistic/sent

Das Element nimmt Statistiken über die von dem Datenlieferanten exportierten Datensätze auf und muss vom Datenlieferanten selbst ausgefüllt werden.

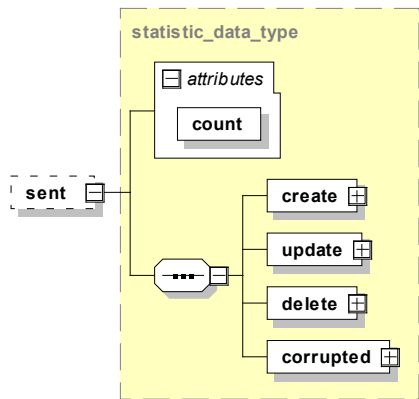


Abbildung 45: Aufbau des Elements sent

Es besteht aus vier Elementen und einem Attribut.

Tabelle 88: Attribut des Elements sent

Name	Type	Use	Beschreibung
count	non_negative_integer_type	required	Gesamtzahl von Vorgängen (Ganze Zahl >= 0): Summe von (<create>+<update>+<delete>+<corrupted>)

Tabelle 89: Kind-Elemente des Elements statistic/sent

Kind-Element	Beschreibung
<create>	Anzahl der Datensätze, die neu importiert werden sollen.
<update>	Anzahl der Datensätze, die aktualisiert werden sollen (z. B. nach einer Korrektur).
<delete>	Anzahl der Datensätze, die von der Datenannahmestelle/Bundesauswertungsstelle storniert werden müssen.
<corrupted>	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 nur die Bundesauswertungsstelle processed bearbeitet. Die Bundesauswertungsstelle schreibt in processed, wie viele Datensätze tatsächlich neu (create), überschrieben (update) und storniert (delete) und ggf. wie viele Datensätze fehlerhaft sind. Außerdem schreibt sie im Attribut processed/@count die Gesamtsumme der verarbeiteten Fälle. Alle Werte sind vom Leistungserbringer mit der Zahl „0“ vorbelegt. Die Angaben im Element sent können als Absichtserklärung der Datenlieferung betrachtet werden. Die Angaben im Element processed sind als Ergebnis der Verarbeitung in Bezug auf den Zieldatenpool zu verstehen.

Leistungserbringer und DAS ignorieren die Angaben in statistic/processed, wenn status_document/@V dem Wert ERROR entspricht. Wenn status_document/@V gleich ERROR ist, dann ist jeder Datensatz zu corrupted zu zählen. Aus technischen Gründen kann nicht sichergestellt werden, dass bei status_document/@V gleich ERROR processed/corrupted/@V geschrieben werden konnte. Daher gilt, wenn status_document/@V gleich ERROR, dann sind alle Fälle corrupted, unabhängig von dem Wert in processed/corrupted/@V.

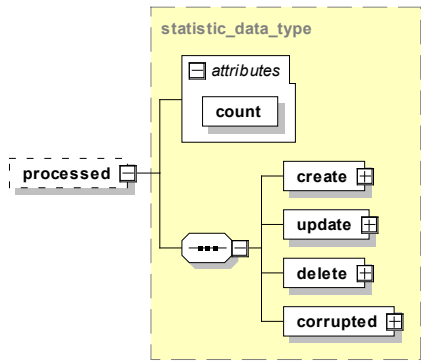


Abbildung 46: Aufbau des Elements processed

Tabelle 90: 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 91: 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.

Tabelle 92: Algorithmus zum Schreiben von statistic/processed in der Bundesauswertungsstelle

DB	gelieferte und geprüfte Exportdatei			Zähler in processed
db_result ⁷⁸	status_document	status_case	<u>./case_admin/action</u>	
n/a (Abfrage wäre überflüssig)	ERROR	OK, WARNING oder ERROR	create, update oderdelete	processed wird nicht durch die Bundesauswertungsstelle editiert und die default-Werte "0" bleiben jeweils unverändert ***
	OK oder WARNING	ERROR	create, update oderdelete	corrupted++ ***
null oder delete	OK oder WARNING	OK oder WARNING	create	create++
			update	create++
			delete	corrupted++ ***
create oder update	OK oder WARNING	OK oder WARNING	create	update++
			update	update++
			delete	delete++
*** case_admin/version/@V ist dennoch zu persistieren, siehe PBDOK.vPurefung.meta_DS				

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.

⁷⁸ db_result = SELECT db.case_admin.action FROM Echtdatenpool db WHERE db.guid = ./case_admin/guid@V and db.status_case != "ERROR" and db.status_document != "ERROR" ORDER BY version DESC LIMIT 1;

```
<qs_data xsi:type="qs_data_17n2_type" module="17/2">
```

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.
- Jeder Teildatensatz besitzt einen Namen, der innerhalb eines Moduls eindeutig ist und unter definierten Bedingungen mehrfach pro Fall erzeugt werden kann.

Folgende Abbildungen zeigen einen einfachen Bogen.

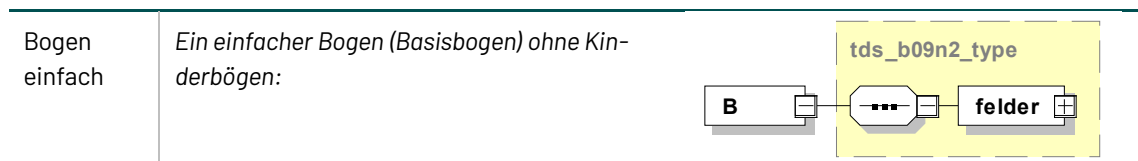


Abbildung 47: Diagramme „Bogen komplex“ und „Bogen einfach“

Die XML-Schemata der einzelnen Module sind in der Schnittstelle „interface_LE“ zu finden:

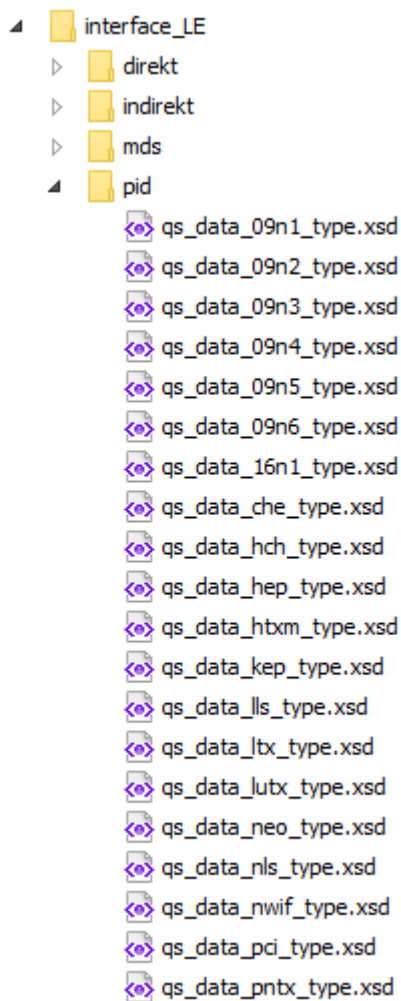


Abbildung 48: Ausprägungen des qs_data-Elements (Erfassungsmodule)

4.5 Umgang mit Umlaut-Domains und E-Mail-Adressen

Die Nutzung von Umlauten ist durch die Internationalized Domain Names (IDN) möglich. Domainnamen wie z. B. <http://www.qualitätsklinikverbund.de/> sind damit nutzbar. Da die Verwendung von Zeichen aus einem Unicode-Zeichensatz in Domainnamen und E-Mail-Adressen immer noch Probleme in Anwendungsprogrammen bereiten, wurde das ASCII Compatible Encoding (ACE) entwickelt, in dem Regeln zur Überführung von Unicode-Zeichen in ASCII-Zeichen festgelegt wurden. Zu jeder Umlautdomain (IDN-Domain) wird parallel immer eine ACE-Domain registriert. Nähere Informationen findet man z. B. bei Wikipedia⁷⁹. Für das oben genannte Beispiel <http://www.qualitätsklinikverbund.de/> würde der ACE-Name xn--qualittsklinikverbund-b2b.de lauten.

⁷⁹ https://de.wikipedia.org/wiki/Internationalisierter_Domainname

E-Mail-Adressen dürfen im lokalen Teil (vor dem @-Zeichen) keine Umlaute enthalten. Die E-Mail-Adresse müller@muster.de wäre nicht möglich. Möglich ist lediglich mueller@muster.de. Es existieren experimentelle Request for Comments (RFCs)⁸⁰, in denen UTF-8-codierte, internationale E-Mail-Adressen ermöglicht werden. Dies wird jedoch nicht flächendeckend im Produktiveinsatz genutzt, aus diesem Grund wird in dieser Spezifikation von der Annahme ausgegangen, dass der lokale Teil der E-Mail-Adresse immer durch ASCII-Zeichen repräsentiert wird.

In den E-Mail-Elementen des XML-Schemas sowie in der Auftragsdatei sind ausschließlich Zeichen des ASCII-Zeichensatzes erlaubt. E-Mail-Adressen mit einem IDN Domainnamen sind entsprechend den ACE-Regeln durch ASCII-Zeichen darzustellen und zu übermitteln.

⁸⁰ <https://tools.ietf.org/html/rfc5336>

5 Tools

Das vorliegende Kapitel beschreibt Spezifikationskomponenten, die als Hilfsprogramme die Prozesse in der Qualitätssicherung unterstützen. Zu den Hilfsprogrammen gehören im Rahmen der Basisspezifikation ein Verschlüsselungsmodul, ein Datenprüfprogramm und ein LE-Pseudonymisierungsprogramm. Das Verschlüsselungspaket und das LE-Pseudonymisierungsprogramm sind jeweils im zugehörigen Anwenderhandbuch und in der dazugehörigen Datenflussspezifikation beschrieben. Das Datenprüfprogramm wird hingegen in den folgenden Abschnitten erklärt.

5.1 Datenprüfprogramm

Das Datenprüfprogramm wird vor dem Hintergrund der Anforderungen der Richtlinie zur datengestützten einrichtungübergreifenden Qualitätssicherung (DeQS-RL) bereitgestellt, um nicht nur beim Leistungserbinger, sondern auch bei den Datenanahmestellen eine Datenprüfung durchzuführen. Das Datenprüfprogramm bezieht die Plausibilitätsregeln direkt aus der Spezifikation und testet Daten vor deren Verschlüsselung.

Das Ausführen vom Datenprüfprogramm setzt Java in der Version 8 oder höher voraus.

5.1.1 Umfang der Prüfungen

Es werden zwei wesentliche Bereiche mit diesem Programm geprüft:

- Schema-Konformität (Struktur)
Die XML-Datei wird dabei auf Konformität mit dem zugrundeliegenden 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.

5.1.2 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>).

5.1.3 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 Dateiendung `.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 50). Eine detaillierte Übersicht über die Anwendung weicher Schemata mit dem Datenprüfprogramm finden sie im Abschnitt B 4.2.



Hinweis

Neben den Leistungserbringern sind die Datenannahmestellen für Krankenhäuser (LKG) sowie für kollektivvertraglich tätige Leistungserbringungen (KV) und die Bundesauswertungsstelle (BAS) die Stellen, die QS-Daten gemäß den G-BA-RL entschlüsseln dürfen und die weiche Variante benötigen.

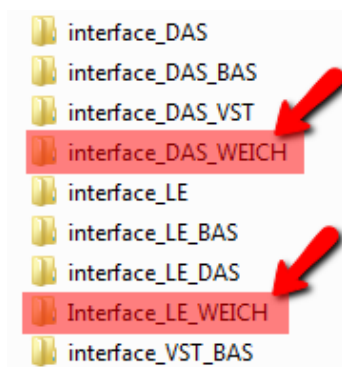


Abbildung 49: Weiche Schemavariante für das DPP

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.

5.1.4 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.

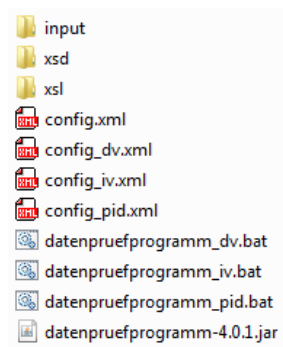


Abbildung 50: Beispiel einer typischen Verzeichnisstruktur

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 51 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.

5.1.5 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 51: Beispiel für eine `Index.html` Datei im Ordner `<output>/html`

5.1.6 Grafische Oberfläche

Wird der Parameter `GUI` in der Konfigurationsdatei auf „true“ gesetzt, wird das Datenprüfprogramm mit einer einfachen grafischen Oberfläche gestartet.

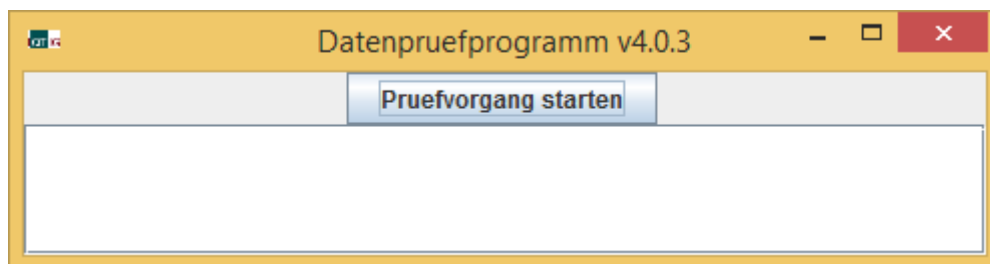


Abbildung 52: Grafische Oberfläche des Datenprüfprogramms

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.

5.1.7 Programmierschnittstelle – API

Sämtliche Funktionen des Datenprüfprogramms können über eine Programmierschnittstelle (API) aufgerufen und direkt in einem Java-Umfeld verwendet werden.

5.2 Verschlüsselungspaket

Das IQTIG stellt ein Ver- und Entschlüsselungspaket für die Anwendung in der externen Qualitätssicherung entsprechend DeQS-RL zur Verfügung. 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. Die Verschlüsselungsprogramme werden separat auf der auf der IQTIG-Website veröffentlicht.

5.3 LE-Pseudonymisierungsprogramm

Das LE-Pseudonymisierungsprogramm richtet sich ausschließlich an die Datenannahmestellen (DAS), um die leistungserbringeridentifizierenden Daten bei den DeQS-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.

C Anhang

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 bestimmbar 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 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
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 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.
Kostenträger	Personen und Institutionen, die die Kosten für medizinische Versorgungsleistungen tragen. Im Rahmen der gesetzlich verankerten Qualitätssicherung sind dies die gesetzlichen Krankenkassen.
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 des entsprechenden QS-Verfahrens 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).

Begriff	Beschreibung
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.
Pseudocode	Programmcodem, der das zugrundeliegende Prinzip eines Algorithmus beschreibt, selbst aber nicht lauffähig ist. Er dient zur Veranschaulichung, unabhängig von der konkret zu verwendenden Programmiersprache.
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).
QS-Auslösung	Initiierung einer Dokumentationspflicht zu Zwecken der Qualitätssicherung (QS-Dokumentation). Bei einer Erhebung vorhandener Daten (z. B. Sozialdaten bei den Krankenkassen) analog das 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
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.

Begriff	Beschreibung
Qualitätssicherung, externe stationäre	Einrichtungsübergreifende Qualitätssicherung für medizinisch-pflegerische Leistungen, die ausschließlich im stationären Sektor erbracht werden.
Qualitätssicherungsmaßnahmen	Strukturierte, in Richtlinien geregelte Vorgehensweise, die Leistungserbringer bei der kontinuierlichen Qualitätsverbesserung unterstützt. Auslöser der Qualitätssicherungsmaßnahmen sind rechnerische Auffälligkeiten im Ergebnis eines Qualitätsindikators.
Regelbetrieb	auch: Routinebetrieb oder Echtbetrieb. Verpflichtende und flächendeckende Umsetzung eines QS-Verfahrens.
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. ambulant/stationär).
Sollstatistik	Aufstellung der im Spezifikationsjahr zu dokumentierenden Leistungen, die vom Krankenhaus durch Konformitätserklärung schriftlich bestätigt wird. Sie gibt die zu erwartende Anzahl von Fällen in den einzelnen QS-Verfahren (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

Begriff	Beschreibung
	QS-Verfahren ist die Spezifikation als das Ergebnis der informations-technischen Aufbereitung zu betrachten.
Spezifikationsjahr	Das Jahr, in dem die Daten erhoben werden. Hierauf beruhen die Ergebnisse der Indikatoren. Die Kriterien für die Abgrenzung des Spezifikationsjahres sind in der Spezifikation zur Sollstatistik definiert. Bezogen auf die Spezifikation meint der Begriff Spezifikationsjahr das Jahr des Aufnahmedatums bei stationärer Behandlung bzw. das Jahr des Behandlungsdatums bei ambulanten Eingriffen der jeweils dokumentationspflichtigen Behandlungsfälle. Wenn stationäre Krankenhaus-Patienten über einen Kalenderjahreswechsel hinweg stationär behandelt werden, kann der Dokumentationsabschluss erst im Folgejahr des Spezifikationsjahres erfolgen.
Stellungnahmeverfahren	Instrument der Qualitätsförderung. Strukturiertes Verfahren, das Einrichtungen bei der kontinuierlichen Verbesserung von Prozessen und Qualität unterstützt. Auslöser des Stellungnahmeverfahrens sind rechnerische Auffälligkeiten im Ergebnis eines Qualitätsindikators im Rahmen der statistischen Basisprüfung der übermittelten QS-Daten.
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 regelt die DeQS-RL sowie die themenspezifischen 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.
Vollzähligkeitsprüfung	Abgleich der laut QS-Filter-Software zu erwartenden Anzahl von Fällen in einem QS-Verfahren (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 QS-Verfahren, die über einen gemeinsamen Dokumentationsbogen erfasst werden, dient der Zählleistungsbereich der Zuordnung zu einer definierten Teilmenge eines QS-Verfahrens.

Impressum

HERAUSGEBER

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

Telefon: (030) 58 58 26-0

info@iqtig.org

iqtig.org