



Institut für Qualitätssicherung und
Transparenz im Gesundheitswesen

Technische Dokumentation zur Spezifikation der einrichtungsbezogenen QS-Dokumentation

Erfassungsjahr 2017

Erstellt im Auftrag des
Gemeinsamen Bundesausschusses

Stand: 7. April 2017

Impressum

Thema:

Technische Dokumentation zur Spezifikation der einrichtungsbezogenen QS-Dokumentation.
QS-Filter, QS-Dokumentation. Erfassungsjahr 2017

Auftraggeber:

Gemeinsamer Bundesausschuss

Datum der Veröffentlichung:

7. April 2017

Herausgeber:

IQTIG – Institut für Qualitätssicherung
und Transparenz im Gesundheitswesen

Katharina-Heinroth-Ufer 1
10787 Berlin

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

info@iqtig.org

<https://www.iqtig.org>

Hinweis:

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

Inhaltsverzeichnis

| | |
|------------------------------------------------------------------------------|----|
| Tabellenverzeichnis | 6 |
| Abbildungsverzeichnis | 7 |
| Abkürzungsverzeichnis..... | 8 |
| Änderungsindex | 11 |
| Leseanleitung..... | 12 |
| 1 Einleitung..... | 13 |
| 1.1 Spezifikationsbegriff..... | 13 |
| 1.1.1 Benennungsschema für Spezifikationspakete..... | 14 |
| 1.1.2 Benennungsschema für Spezifikationskomponenten..... | 15 |
| 1.2 Zielsetzung und Zielgruppe | 15 |
| 1.3 Releaseplanung..... | 16 |
| A Prozesse..... | 17 |
| Auslösung | 17 |
| Erfassung..... | 17 |
| Export der Daten | 19 |
| Datenübermittlung..... | 20 |
| Rückprotokollierung | 20 |
| B Komponenten..... | 21 |
| 1 QS-Filter..... | 24 |
| 1.1 Anmerkungen zur Struktur der Spezifikationsdatenbank für QS-Filter | 24 |
| 1.2 Grundlegende Tabellen der Datenbank | 24 |
| 1.2.1 Module (Datensätze der QS-Dokumentation)..... | 24 |
| 1.2.2 Struktur der Datensatzdefinitionen | 25 |
| 1.2.3 OPS-Listen | 29 |
| 1.2.4 ICD-Listen | 29 |
| 1.2.5 EBM-Listen..... | 30 |
| 1.2.6 Entgelt-Listen..... | 31 |
| 1.2.7 Versionsverwaltung..... | 31 |
| 1.2.8 Meta-Tabellen..... | 32 |
| 1.2.9 DB-Änderungen gegenüber der Vorgängerversion..... | 32 |

| | | |
|-------|----------------------------------------------------------------------|----|
| 1.3 | Der QS-Filter-Datensatz..... | 32 |
| 1.3.1 | Der QS-Filter-Eingangsdatensatz | 33 |
| 1.4 | Der Algorithmus zur Ermittlung der Dokumentationspflicht..... | 39 |
| 1.4.1 | Einleitung und Überblick..... | 39 |
| 1.4.2 | Leistungsbereichsbezogene Einschlusskriterien..... | 40 |
| 1.4.3 | Administrative Einschlusskriterien..... | 42 |
| 1.4.4 | Struktur und Syntax der Auslösebedingungen..... | 43 |
| 1.4.5 | Fehlerprüfung..... | 46 |
| 2 | QS-Dokumentation..... | 49 |
| 2.1 | Anmerkungen zur Struktur der Spezifikation zur QS-Dokumentation..... | 49 |
| 2.2 | Datenfeldbeschreibung | 49 |
| 2.2.1 | Dokumentationsmodule (Datensätze) | 51 |
| 2.2.2 | Teildatensätze..... | 52 |
| 2.2.3 | Datenfelder (Bogenfelder) | 53 |
| 2.2.4 | Überschriften..... | 60 |
| 2.2.5 | Ausfüllhinweise..... | 60 |
| 2.3 | Plausibilitätsprüfungen..... | 61 |
| 2.3.1 | Die Regeltabelle | 62 |
| 2.3.2 | Regelsyntax..... | 63 |
| 2.3.3 | Funktionen | 68 |
| 2.3.4 | Syntaxvariablen..... | 69 |
| 2.3.5 | Einzelregeln | 69 |
| 2.3.6 | Teildatensatzübergreifende Regeln | 70 |
| 2.3.7 | Feldgruppenregeln..... | 70 |
| 2.3.8 | Prüfung von Feldeigenschaften..... | 76 |
| 2.3.9 | Verfahren für die Evaluation von Regeln..... | 79 |
| 2.4 | Exportfeldbeschreibung..... | 80 |
| 2.4.1 | Exportmodule | 80 |
| 2.4.2 | Exportdatensatz..... | 81 |
| 2.5 | Versionierung..... | 81 |
| 2.5.1 | Grundlegende Definitionen..... | 81 |
| 2.5.2 | Delta-Informationen zur vorhergehenden Version..... | 82 |

| | | |
|-----|-----------------------------|----|
| 2.6 | Administrative Objekte..... | 83 |
| 3 | XML-Schema..... | 84 |
| 4 | Tools..... | 85 |
| C | Anhang..... | 86 |
| | Glossar..... | 86 |

Tabellenverzeichnis

| | |
|-------------------------------------------------------------------------|----|
| Tabelle 1: Struktur der Tabelle Ds..... | 25 |
| Tabelle 2: Struktur der Tabelle Tds..... | 26 |
| Tabelle 3: Struktur der Tabelle BasisTyp..... | 27 |
| Tabelle 4: Struktur der Tabelle Schluessel | 28 |
| Tabelle 5: Struktur der Tabelle SchluesselWert..... | 29 |
| Tabelle 6: Struktur der Tabelle Version | 31 |
| Tabelle 7: Ausschnitt der Tabelle Ds..... | 33 |
| Tabelle 8: Felder des QS-Filter-Eingangsdatensatzes nach §301 | 34 |
| Tabelle 9: Felder des QS-Filter-Eingangsdatensatzes nach § 295 | 38 |
| Tabelle 10: Struktur der Tabelle ModulAusloeser..... | 41 |
| Tabelle 11: Struktur der Tabelle AdminKriterium..... | 43 |
| Tabelle 12: Basistypen der Variablen | 43 |
| Tabelle 13: Präzedenz und Assoziativität der Operatoren..... | 45 |
| Tabelle 14: Fehlercodes des QS-Filters..... | 46 |
| Tabelle 15: Struktur der Tabelle Modul..... | 51 |
| Tabelle 16: Inhalte der Tabelle BogenTyp..... | 53 |
| Tabelle 17: Struktur der Tabelle BogenFeld..... | 54 |
| Tabelle 18: Struktur der Tabelle Feld..... | 56 |
| Tabelle 19: Struktur der Tabelle BasisTyp..... | 57 |
| Tabelle 20: Struktur der Tabelle Schluessel..... | 58 |
| Tabelle 21: Struktur der Tabelle SchluesselWert..... | 59 |
| Tabelle 22: Struktur der Tabelle Abschnitt | 60 |
| Tabelle 23: Arten von Hinweistypen..... | 61 |
| Tabelle 24: Tabelle RegelTyp..... | 61 |
| Tabelle 25: Struktur der Tabelle Regeln..... | 62 |
| Tabelle 26: Struktur der Tabelle RegelFelder | 63 |
| Tabelle 27: Basistypen der Datenfelder in den Plausibilitätsregeln..... | 64 |
| Tabelle 28: Präzedenz und Assoziativität der Operatoren..... | 65 |
| Tabelle 29: Typen von Feldgruppen..... | 71 |
| Tabelle 30: Struktur der Tabelle FeldGruppe..... | 72 |
| Tabelle 31: Struktur der Tabelle FeldgruppeFelder..... | 73 |
| Tabelle 32: Formale Definition einer Feldgruppe | 74 |
| Tabelle 33: Struktur der Tabelle ExportModul..... | 81 |

Abbildungsverzeichnis

| | |
|-----------------------------------------------------------------------------------------------|----|
| Abbildung 1: Serielles Datenflussmodell für die sektorenübergreifende Qualitätssicherung..... | 19 |
| Abbildung 2: Tabellen und Relationen der Datenfeldbeschreibung..... | 50 |
| Abbildung 3: Algorithmus zur Evaluation von Plausibilitätsregeln | 80 |

Abkürzungsverzeichnis

| Abkürzung | Bedeutung |
|------------|------------------------------------------------------------------------------------------------------------------------------------|
| AG | Arbeitsgruppe |
| AG Qesü-RL | Arbeitsgruppe Qesü-RL des G-BA |
| BAS | Bundesauswertungsstelle |
| BE | Bundesebene |
| BSNR | Betriebsstättennummer |
| DAS | Datenannahmestelle |
| DAS-SV | Datenannahmestellen für selektivvertraglich erbrachte Leistungen |
| DIMDI | Deutsches Institut für Medizinische Dokumentation und Information |
| DRG | Diagnosis Related Groups (diagnosebezogene Fallgruppen) |
| EBM | Einheitlicher Bewertungsmaßstab |
| FU | Follow-up |
| G-BA | Gemeinsamer Bundesausschuss |
| GKV | Gesetzliche Krankenversicherung |
| GOP | Gebührenordnungsposition |
| GUID | Globally Unique Identifier |
| ICD | International Classification of Diseases (Internationale 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 | Institutionskennzeichen |
| iv | Indirekte Verfahren |
| KBV | Kassenärztliche Bundesvereinigung |
| KH | Krankenhaus |
| KHEntgG | Gesetz über die Entgelte für voll- und teilstationäre Krankenhausleistungen (Krankenhausentgeltgesetz) |
| KIS | Krankenhausinformationssystem |

| Abkürzung | Bedeutung |
|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| KV | Kassenärztliche Vereinigung |
| KVDT | Kassenärztliche Vereinigung-Datentransfer (Datenformat) |
| LANR | Lebenslange Arztnummer |
| LE | Leistungserbringer |
| LKG | Landeskrankenhausgesellschaft(en) |
| LQS | Landesgeschäftsstelle für Qualitätssicherung |
| MDS | Minimaldatensatz |
| MVZ | Medizinisches Versorgungszentrum |
| OPS | Operationen- und Prozedurenschlüssel |
| OR | ODER-Operator |
| PB | Probetrieb |
| PID | Patientenidentifizierende Daten |
| PR | Arztpraxis |
| Qesü-RL | Richtlinie (des GBA) 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 |
| RAM | Random Access Memory (Arbeitsspeicher) |
| RL | Richtlinie |
| RSA | Verfahren zur Datenverschlüsselung, entwickelt von R. Rivest, A. Shamir und L. Adleman |
| SGB | Sozialgesetzbuch |
| SGB V | Sozialgesetzbuch Fünftes Buch |
| SQG | Sektorenübergreifende Qualität im Gesundheitswesen |
| SWA | Softwareanbieter |
| TB | Testbetrieb |
| TDS | Teildatensatz |
| 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 |

| Abkürzung | Bedeutung |
|------------------|----------------------------------------------------------------------------------------------------------|
| XSLT | Extensible Stylesheet Language Transformation (Programmiersprache zur Transformation von XML-Dokumenten) |

Änderungsindex

Änderungen der Datenbanken im Vergleich zur Vorversion lassen sich anhand der Delta-Tabellen nachvollziehen. Da bei der vorliegenden Spezifikation der einrichtungsbezogenen QS-Dokumentation kein Vergleich zu einer Vorversion hergestellt werden kann, sind in den Delta-Tabellen der Version 2017 V01 Alpha keine Änderungen ausgewiesen.

| Änderung | Kapitel/Abschnitt | Version |
|---------------------------------------------------------|--------------------------|----------------|
| Beschreibung der einrichtungsbezogenen QS-Dokumentation | - | 2017 V01 Alpha |

Leseanleitung

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

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

Legende

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



Achtung

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



Hinweis

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

Beispiel:

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

1 Einleitung

Die vorliegende technische Dokumentation beschreibt die Spezifikation zur einrichtungs- und sektorenübergreifenden Qualitätssicherung gemäß §§136 ff. SGB V.

Die Spezifikation der einrichtungsbezogene QS-Dokumentation umfasst alle Komponenten im Zusammenhang mit der Datenerfassung, d.h. von der Bestimmung einer Dokumentationspflicht bis hin zur Rückprotokollierung übermittelter Datensätze. Die vorliegende technische Dokumentation beschreibt die hierbei relevanten Prozesse und Komponenten.

Regelungsbereich der Spezifikation der einrichtungsbezogenen QS-Dokumentation ist das Verfahren *Vermeidung nosokomialer Infektionen – postoperative Wundinfektionen* gemäß der Richtlinie zur einrichtungs- und sektorenübergreifenden Qualitätssicherung (Qesü-RL)¹, die vom Gemeinsamen Bundesausschuss (G-BA) beschlossen wird. Im Erfassungsjahr 2017 beinhaltet die Spezifikation der einrichtungsbezogenen QS-Dokumentation daher Module des Verfahrens *Vermeidung nosokomialer Infektionen – postoperative Wundinfektionen*.

Die Spezifikation der einrichtungsbezogenen QS-Dokumentation bezieht sich dabei auf die QS-Dokumentation beim Leistungserbringer – für eine Nutzung von Sozialdaten bei den Krankenkassen gemäß §299 SGB V wird eine unabhängige Spezifikation veröffentlicht.² Für eine spezifikationskonforme Umsetzung von QS-Softwareprodukten sind alle Spezifikationskomponenten zu berücksichtigen.

1.1 Spezifikationsbegriff

Die Spezifikation der einrichtungsbezogenen QS-Dokumentation ist die Gesamtheit aller Vorgaben, nach denen die Bestimmung der dokumentationspflichtigen Einrichtungen, die QS-Dokumentation selbst und die Übermittlung der Daten erfolgen sollen, bezogen auf ein Erfassungsjahr. Die Dokumentationspflicht einer Einrichtung wird ausgelöst, wenn mindestens ein Fall den definierten Auslösekriterien entspricht. Die Zuordnung eines Falles zu einer Spezifikation richtet sich nach dem Kalenderjahr der Aufnahme in das Krankenhaus bzw. dem Eingriffsdatum bei ambulanten Fällen.

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

Damit gibt es ein Spezifikationspaket für

- den Regelbetrieb (QSKH und Qesü)

¹ <https://www.g-ba.de/informationen/richtlinien/72/>

² <https://iqtig.org/datenservice>

- ggf. Sonderexporte
- ggf. Probetriebe
- ggf. Testbetriebe

Sowohl die Spezifikationspakete als auch die einzelnen Komponenten werden nach einem einheitlichen Schema benannt, das bereits im Namen übersichtlich die relevanten Informationen wie Betriebsart, Exportformat und Versionierung enthält. Dieses Schema wird im nächsten Abschnitt detailliert erläutert. Durch die Versionierung sowohl auf der Ebene der Pakete als auch auf der Ebene der Komponenten ist gewährleistet, dass der aktuelle Stand leicht ersichtlich ist. Zudem wird die Kommunikation über die anzuwendenden Bestandteile der Spezifikation erleichtert.

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

1.1.1 Benennungsschema für Spezifikationspakete

Die Benennung der Spezifikationspakete setzt sich wie folgt zusammen:

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

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

Da die Spezifikation der einrichtungsbezogenen QS-Dokumentation im Erfassungsjahr 2017 auf das Verfahren 2 (QS WI) der Richtlinie zur einrichtungs- und sektorenübergreifenden Qualitätssicherung beschränkt ist, steht im Platzhalter <Richtlinie> immer die Bezeichnung Qesue.

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

- FDOK: fallbezogene QS-Dokumentation
- EDOK: einrichtungsbezogene QS-Dokumentation
- SozDat: Nutzung von Sozialdaten bei den Krankenkassen
- DBSD: Datenformat zum Bericht über den Strukturierten Dialog

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

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

Bei der Betriebsart kann zwischen folgenden Kürzeln unterschieden werden:

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

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

Beispiele:

Die einrichtungsbezogene QS-Dokumentation kann beispielsweise in folgendem Spezifikationspaket veröffentlicht werden:

2017_Qesue_EDOK_RB_XML_V01

Die einrichtungsbezogene QS-Dokumentation (Dokumentation von Einrichtungen, die ambulante und stationäre Fälle am Krankenhaus behandeln) kann beispielsweise in folgendem Spezifikationspaket veröffentlicht werden:

2017_Qesue_EDOK_LKG_RB_XML_V01

Die einrichtungsbezogene QS-Dokumentation (Dokumentation von Einrichtungen, die durch die KV abgerechnete Fälle behandeln) kann beispielsweise in folgendem Spezifikationspaket veröffentlicht werden:

2017_Qesue_EDOK_KV_RB_XML_V01

1.1.2 Benennungsschema für Spezifikationskomponenten

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

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

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

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

Die Bezeichnungen der einzelnen Komponenten werden in Abschnitt B Komponenten erläutert.

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. Dadurch werden die Bereitstellung valider und vergleichbarer Daten sowie ein unter datenschutzrechtlichen Gesichtspunkten sicherer Datenfluss gewährleistet.

1.3 Releaseplanung

Um Planungssicherheit zu gewährleisten und angemessen auf Fehler reagieren zu können, werden die Termine zur Veröffentlichung von Spezifikationspaketen (Versionen) eines Erfassungsjahres und zu Rückmeldefristen in der Technischen Dokumentation angekündigt. Die Releaseplanung der einrichtungsbezogenen QS-Dokumentation ist an die QS-Basispezifikation für Leistungserbringer angelehnt, die in der Technischen Dokumentation zur Basispezifikation für Leistungserbringer beschrieben ist. Die dort genannten Meilensteine sind als Zielwerte zu betrachten und werden im Regelbetrieb entsprechend der Vorgaben des G-BA, in Abstimmungen mit Verfahrensteilnehmern sowie Anforderungen aus der Umsetzung konkretisiert. Es erfolgt eine regelmäßige Abstimmung mit Verfahrensteilnehmern (z.B. Softwareherstellern, Datenannahmestellen, Vertrauensstelle) in Form von Informationstreffen, Workshops und Kommunikationsplattform. Zudem wurden neue Meilensteine in den Prozess der Systempflege integriert (z.B. die Bereitstellung von Änderungsempfehlungen und Alphaversionen) und Festlegungen getroffen, die die Qualität der Spezifikation erhöhen und die Richtlinienkonformität sicherstellen (z.B. werden wesentliche Änderungen nur im Rahmen finaler Versionen berücksichtigt).



Hinweis

An dieser Stelle werden bis zur Veröffentlichung der finalen Version entsprechende Termine zur Veröffentlichung von Spezifikationspaketen beschrieben.

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 mitgeteilt werden. Sollten Sie keine Zugangsdaten zur Kommunikationsplattform haben, obwohl Sie eine beteiligte Institution (z.B. Softwareanbieter, Datenannahmestelle) sind, lassen Sie sich bitte bei uns registrieren.

Ihr Ansprechpartner:

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

Telefon: (+49) 30 58 58 26 340

Fax: (+49) 30 58 58 26 341

verfahrenssupport@iqtig.org

www.iqtig.org

A Prozesse

Im Folgenden werden die einzelnen Prozesse und Unterprozesse der Spezifikation beschrieben.

Auslösung

Für jeden Fall im Krankenhaus oder in der Arztpraxis wird geprüft, welche Datensätze (Module) der externen Qualitätssicherung dokumentationspflichtig sind. Entspricht mindestens ein Fall (Tracer-Operation) den definierten Auslösekriterien, wird die einrichtungsbezogene QS-Dokumentation ausgelöst. Mit der vorliegenden Spezifikation können Systeme entwickelt werden, die eine zeit- und prozessnahe Auslösung von QS-Dokumentationen ermöglichen.

Der Algorithmus (Abschnitt B 1.4) trifft seine Entscheidung auf der Grundlage der medizinischen Routinedokumentationen bzw. Abrechnungsdaten (Kodierung der Diagnosen (Haupt- und Nebendiagnosen, Primär- und Sekundärdiagnosen), Prozeduren, Gebührenordnungspositionen gemäß EBM-Katalog).

Einrichtungen, die ambulante und stationäre Fälle am Krankenhaus behandeln

Die medizinischen Routinedokumentationen sind in jedem Krankenhausinformationssystem (KIS) 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.

Einrichtungen, die ambulante vertragsärztliche oder stationäre belegärztliche Leistungen erbringen

Für diese Einrichtungen (z.B. Arztpraxen) ist die Dokumentationspflicht über die bei den Kassenärztlichen Vereinigungen vorliegenden Abrechnungsdaten zu bestimmen. Im QS-Filter-Eingangsdatensatz (Abschnitt B 1.3) ist definiert, welche Daten verwendet werden. Diese basieren auf der KVDT-Datensatzbeschreibung (ungeprüfter Eingangsdatensatz) der Kassenärztlichen Bundesvereinigung für den Einsatz von IT-Systemen in der Arztpraxis zum Zwecke der Abrechnung gemäß §295 Abs. 4 SGB V.

Erfassung

Nach Auslösung eines Moduls durch den QS-Filter folgt der Prozess der Erfassung. Neben QS-Daten sind auch einrichtungsidentifizierende Daten wie das Institutionskennzeichen für das Krankenhaus oder die Betriebsstättennummer des niedergelassenen Leistungserbringers zu dokumentieren. Die Erfassung ist abgeschlossen, wenn alle zu dokumentierenden Datenfelder unter Berücksichtigung von Abhängigkeiten und Plausibilitätsprüfungen vollständig erfasst wurden. Die Regelungen zur Gestaltung von Eingabemasken sind der im Erfassungsjahr gültigen Version der Technischen Dokumentation zur Basisspezifikation für Leistungserbringer zu entnehmen.

Plausibilitätsprüfungen

Fehlende und widersprüchliche Angaben in den Datensätzen sollen durch umfangreiche Plausibilitätsprüfungen verhindert werden. In der QS-Dokumentation 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 weitgehend entfällt.

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 folgenden Arten von Plausibilitätsprüfungen unterschieden:

- **harte Prüfungen:** Bei einer harten Regelverletzung ist ein Dokumentationsabschluss der QS-Dokumentation eines Vorgangs unzulässig. Die in der technischen Dokumentation und der Datenbank definierten Plausibilitätsprüfungen sind hart, außer wenn sie explizit als weich gekennzeichnet sind.
- **weiche Prüfungen:** Die weichen Plausibilitätsprüfungen der Spezifikation sind bis spätestens zum Dokumentationsabschluss durchzuführen. Bei einer Regelverletzung erhält der Benutzer einen Warnhinweis, anhand dessen er entscheidet, ob eine Änderung von Feldinhalten notwendig ist.

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 2.3.8 zu entnehmen.

Es müssen sowohl die harten als auch die weichen feldbezogenen Prüfungen ausgeführt werden. 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 Schlüsselkodes.

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 2.3.7 erläutert.

Export der Daten

Die dokumentierten QS-Daten müssen bei sektorenübergreifenden Verfahren gem. Qesü-RL in Abhängigkeit vom Abrechnungskontext der einzelnen Fälle verschiedenen Datenflüssen zugeordnet werden.

Abbildung 1 stellt das serielle Datenflussmodell gem. Qesü-RL dar. Hierbei sind je nach Leistungserbringer bzw. Abrechnungskontext unterschiedliche Datenannahmestellen vorgesehen:

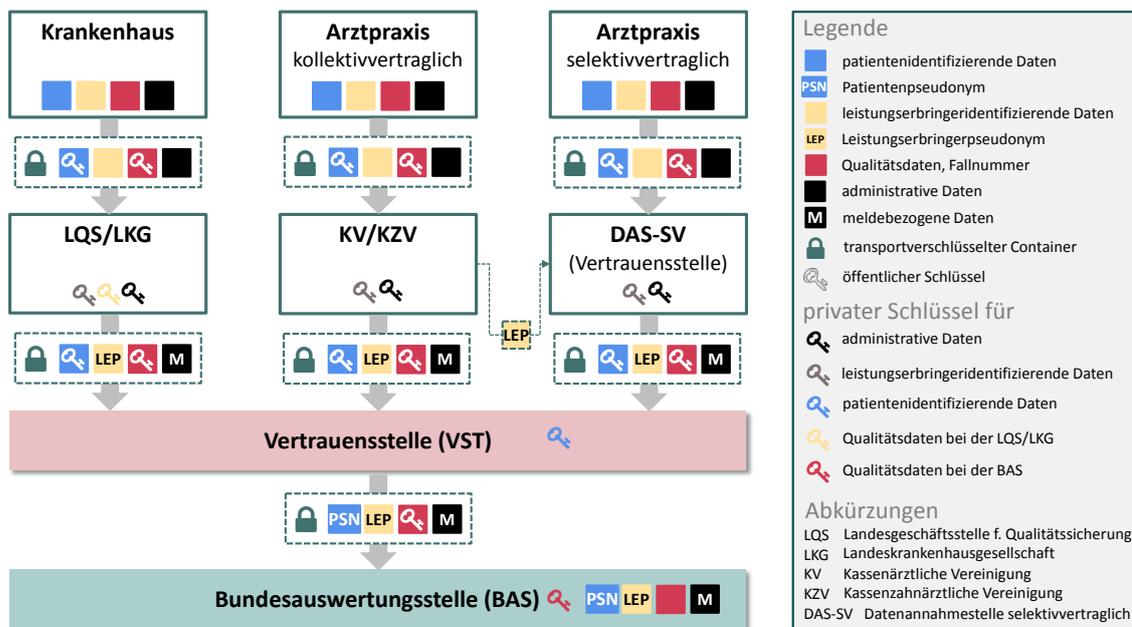


Abbildung 1: Serielles Datenflussmodell für die sektorenübergreifende Qualitätssicherung

Die Zuordnung der einzelnen Fälle zu den Leistungserbringern gemäß Datenfluss nach Qesü-RL richtet sich primär nach dem Abrechnungskontext und nicht nach dem Ort der Leistungserbringung.



Hinweis

An dieser Stelle wird bis zur Veröffentlichung der finalen Version eine entsprechende Dokumentation erstellt.

Datenübermittlung



Hinweis

An dieser Stelle wird bis zur Veröffentlichung der finalen Version eine entsprechende Dokumentation erstellt.

Rückprotokollierung



Hinweis

An dieser Stelle wird bis zur Veröffentlichung der finalen Version eine entsprechende Dokumentation erstellt.

B Komponenten

In diesem Kapitel werden die einzelnen Komponenten der Spezifikation beschrieben. Ein Spezifikationspaket bildet die Gesamtheit seiner Spezifikationskomponenten ab. Mögliche Spezifikationskomponenten sind der Technischen Dokumentation zur Basisspezifikation für Leistungserbringer zu entnehmen.

Spezifikationsdatenbanken

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

Datenbank zur QS-Dokumentation (QSDOK)

Die Datenbank zur QS-Dokumentation dient der Spezifikation von Datenerhebung und -erfassung unter Berücksichtigung von Plausibilitätsprüfungen und zu exportierenden Datenfeldern. Einige wichtige Datenbankinhalte werden über die Dokumentationsbögen (Dokubögen) abgebildet.

Folgende Spezifikationskomponente ergänzt die Datenbank um Hinweise und Informationen:

- Die Ausfüllhinweise (Ausfüllhinweise) dienen als Hilfestellung bei der Dokumentation durch den Anwender. Die Namen der HTML-Dateien für einzelne Datenfelder sind in der Datenbank (BogenFeld.ahinweis) hinterlegt.

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. 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.

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 Attribut³, welches durch Setzen eines weiteren, eindeutigen Indexes (bestehend aus einem oder mehreren Attributen) definiert ist.

Beispiele:

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

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

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

³ Oder sie enthalten eine identifizierende Attributkombination, die einen eindeutigen Schlüssel definiert.

N-M-Beziehungen werden wie üblich über Verknüpfungstabellen realisiert. In der Spezifikation haben Verknüpfungstabellen gewöhnlich keinen Primärschlüssel⁴, 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`.

⁴ 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 Auslösekriterien werden für jeden Fall geprüft. Sind die Kriterien für mindestens einen Fall erfüllt, gilt das jeweilige Modul der externen Qualitätssicherung (NWIES und/oder NWIEA) für eine Einrichtung als dokumentationspflichtig.

1.1 Anmerkungen zur Struktur der Spezifikationsdatenbank für QS-Filter

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

<Erfassungsjahr>_QESUE_QSF_V<Versionsnummer>.mdb

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

Beispiel:

Im Erfassungsjahr 2017 ist die QS-Filter-Spezifikation 2017_QESUE_QSF_V01.mdb⁵ gültig. Die Kennung 2017 gilt für das Erfassungsjahr 2017.

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

Die Tabellenstruktur der Spezifikationsdatenbank wird in Abschnitt B Komponenten beschrieben.

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 die QS-Dokumentation. Diese Datensätze bilden in der Spezifikation für QS-Dokumentation den auszulösenden Dokumentationsbogen (z.B. der Software) ab.

⁵ Die Versionsnummer der gültigen Spezifikation (z.B. V01, V02, usw.) ist dem zuletzt veröffentlichten Update zu entnehmen.

Beispiele:

Einrichtungen, die ambulante und stationäre Fälle am Krankenhaus behandeln

- Der Modulauslöser `NWIES_LKG` löst den QS-Datensatz `NWIES` aus.
- Der Modulauslöser `NWIEA_LKG` löst den QS-Datensatz `NWIEA` aus.

Einrichtungen, die ambulante vertragsärztliche oder stationäre belegärztliche Leistungen erbringen

- Der Modulauslöser `NWIES_KV` löst den QS-Datensatz `NWIES` aus.
- Der Modulauslöser `NWIEA_KV` löst den QS-Datensatz `NWIEA` aus.

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.

Die Struktur der Tabelle `Modul` wird in der Technischen Dokumentation für Leistungserbringer beschrieben.

1.2.2 Struktur der Datensatzdefinitionen

Die Module werden über den definierten Modulauslöser als dokumentationspflichtig erkannt, indem die hinterlegte Bedingung mit den im KIS oder in den Abrechnungsdaten der Kassenärztlichen Vereinigungen gespeicherten Daten geprüft wird. Ist die Bedingung erfüllt, wird das Modul ausgelöst. Da die in der Bedingung enthaltenen Felder im KIS oder in den Abrechnungsdaten der Kassenärztlichen Vereinigungen vorliegen müssen, ist der Eingangsdatensatz 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 die QS-Dokumentation 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

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

Tabelle 1: Struktur der Tabelle `Ds`

| Feldname | Datentyp | Bemerkung |
|-------------------|----------|---------------------------------------------------------------|
| <code>idDs</code> | INTEGER | Primärschlüssel |
| <code>name</code> | TEXT | Technischer Name des Datensatzes (Identifizierendes Attribut) |

| Feldname | Datentyp | Bemerkung |
|-------------|----------|---------------------|
| bezeichnung | TEXT | Beschreibender Text |

Teildatensätze

Die Definition von Teildatensätzen befindet sich in der Tabelle Tds der Datenbank (Tabelle 2). Jeder Teildatensatz ist eindeutig durch seinen Namen (z.B. FALL, PROZ oder DIAG) charakterisiert.

Tabelle 2: Struktur der Tabelle Tds

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

In der Tabelle Tds ist eine Hierarchie der Teildatensätze definiert. Der Ausgangspunkt („root“) für die Teildatensatzhierarchie eines Datensatzes (z.B. QS-Filter-Datensatz) ist immer der Basisteildatensatz (daraus folgt: Teildatensatz⁶ fkTds = NULL in der Tabelle Tds). Über die in den restlichen Teildatensätzen des Moduls definierten Bezüge zu den Mutterteildatensätzen und Relationstypen lässt sich ein Hierarchiebaum der Teildatensätze aufbauen.

Jeder Datensatz besteht aus

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

⁶ Es darf nur eine Definition eines Basisteildatensatzes existieren.

Beispiele:

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

Felder der Teildatensätze

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

Felder

Ein Feld wird eindeutig über seinen technischen Namen definiert. Jedes Feld hat einen Basistyp (z.B. SCHLUESSEL, NUMSCHLUESSEL, ZAHL). Schlüsselfelder erfordern zusätzlich einen Schlüssel (siehe Tabelle 3).

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 3: Struktur der Tabelle `BasisTyp`

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

**Hinweis**

In Zeichenketten (Basistyp `TEXT`) sind alle Zeichen des ASCII-Formats mit einem Kode `> 32` erlaubt. Ausgenommen sind das Semikolon, die doppelten Anführungsstriche und Hochkommata.

- Es gibt zwei Arten von Schlüsseln: numerische und nichtnumerische.
- Das Komma trennt die Nachkommastellen, Vorzeichen `+` und `-` sind erlaubt.
- Das Datumstrennzeichen ist der Punkt.

Schlüssel

Identifizierendes Merkmal eines Schlüssels ist sein technischer Name. Die meisten Schlüsselcodes sind in der Tabelle `SchluesseleWert` (Tabelle 4) definiert. Externe Schlüsselkataloge (z.B. OPS, EBM oder ICD10) sind von den entsprechenden Anbietern zu beziehen.

Tabelle 4: Struktur der Tabelle `Schluessele`

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

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

Beispiel:

Attribut **zahl** bei Schlüsselfeldern

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

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

Externe Schlüsselkataloge

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

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

**Achtung**

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

Schlüsselwerte

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

Tabelle 5: Struktur der Tabelle `SchlüsselWert`

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

**Hinweis**

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

1.2.3 OPS-Listen

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

**Achtung**

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

1.2.4 ICD-Listen

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



Achtung

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

ICD-Kodes mit dem Zusatzkennzeichen A (Ausschluss) sind bei der Auslösung nicht zu berücksichtigen. Bei der Auslösung von ambulanten Fällen ist die Auslösung nur mithilfe von gesicherten Diagnosen zulässig.



Achtung

Auslösung

Bei der Auslösung sind nur gesicherte Diagnosen einzubeziehen.

Bei einer prozessbegleitenden Auslösung über eine QS-Filtersoftware ist sicherzustellen, dass Aktualisierungen im KIS zwischen der Anwendungssoftware und der QS-Dokumentationssoftware kommuniziert werden. Grundsätzlich ist hier eine Übereinstimmung aller Diagnosen mit der Entlassungsanzeige des Behandlungsfalles sicherzustellen.

ICD-Kodes mit dem Zusatzkennzeichen A (Ausschluss) sind bei der Prüfung auf Ausschlusslisten nicht zu berücksichtigen.

1.2.5 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.

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 sind die EBM-Kodes in Tabelle `EBMWert` fünfstellig – ohne ergänzenden Zeichen – definiert. Für die Prüfung, ob zwei Codes identisch sind, genügt kein einfacher Stringvergleich. Stattdessen wird ein Stringvergleich der Normcodes⁷ durchgeführt, um die Übereinstimmung zwischen dem dokumentierten Code und dem einer EBM-Liste zu ermitteln.

⁷ Jeder EBM-Kode lässt sich entweder als Code mit ergänzenden Zeichen (Normcode + ergänzende Zeichen) oder als Code ohne ergänzende Zeichen (Normcode) darstellen.

1.2.6 Entgelt-Listen

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

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

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

Beispiel:

```
STAT_BELEG_ENTGELT    belegärztliche Fälle (FALLART = stationär; gemäß [ENTGEL-
                        TARTSTAT])
```

1.2.7 Versionsverwaltung

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

Die wichtigsten Eigenschaften einer Version sind der Versionsname (Attribut `name`) und der Gültigkeitszeitraum (Attribute `ab` und `bis`).



Achtung

Der Gültigkeitszeitraum einer Version ist in der fallbezogenen QS-Basispezifikation für Leistungserbringer in der Regel ein Erfassungsjahr (z.B. Aufnahme zwischen dem 01.01.2017 und dem 31.12.2017). Die Auslösung der einrichtungsbezogenen QS-Dokumentation bezieht sich bei den Modulen `NWIEA` und `NWIES` nur auf **die ersten beiden Quartale des Erfassungsjahres**.

Die Spezifikation der einrichtungsbezogenen QS-Dokumentation ist beispielsweise für das Erfassungsjahr 2017 vom 01.01.2017 bis zum 31.06.2017 gültig. Die Prozesse des Verfahrens können außerhalb dieses Zeitraums liegen.

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 Versionen⁸: Welche Vorgängerversion vorher gültig war, kann über das Attribut `fkVersion` ermittelt werden.

Tabelle 6: Struktur der Tabelle `Version`

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

⁸ Die Inhalte der Vorversionen sind nicht Teil der aktuellen Spezifikationsdatenbank.

| Feldname | Datentyp | Bemerkung |
|--------------|----------|------------------------------------------------------------------------------------------------------------------------------------------|
| bezeichnung | TEXT | Beschreibender Text |
| ab | DATUM | Anfang des Gültigkeitszeitraumes |
| bis | DATUM | Ende des Gültigkeitszeitraumes |
| pub | DATUM | Datum der Publikation |
| gueltig | BOOLEAN | gültige Version (nur ein Eintrag darf als gültig markiert sein) |
| fkVersion | INTEGER | Bezug zur Vorgängerversion |
| fkVersStatus | CHAR | Bezug zum Status einer Version (Tabelle VersStatus): E = in Entwicklung F = finale Version 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 Systempflege⁹ 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.

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 2.5.2).

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 2.5.2).

1.3 Der QS-Filter-Datensatz

Verbindlich für ein Erfassungsjahr ist die Datenfeldbeschreibung, welche in der gültigen Spezifikationsdatenbank definiert ist. Hier werden die Datensätze `DATENSATZ_301` und `DATENSATZ_295` dargestellt. Beide bilden jeweils den entsprechenden QS-Filter-Eingangsdatensatz ab (Tabelle 7).

⁹ Die Aktualisierung der ICD- und OPS-Kodes erfolgt auch für nicht verpflichtende Module weiterhin im Rahmen des DIMDI-Umstiegs.

Tabelle 7: Ausschnitt der Tabelle Ds

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

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

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

1.3.1 Der QS-Filter-Eingangsdatensatz

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

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

Die Dokumentationspflicht einer Einrichtung wird ausgelöst, wenn mindestens ein Fall des Auslösebedingungen entspricht. Die für den QS-Filter maßgeblichen administrativen und medizinischen Datenfelder zur Identifikation dokumentationspflichtiger Fälle im Rahmen der Qesü-RL für durch ein Krankenhaus erbrachte Fälle finden sich im Aufnahmedatensatz bzw. in der Entlassungsanzeige des §301-Datenformats¹⁰. 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 PAT-ALTER, 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 fünf 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)

¹⁰ 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 ¹¹ wieder. Der QS-Filter-Eingangsdatensatz basiert zum größten Teil auf einer Teilmenge der Inhalte dieser Datensätze.

Tabelle 8: Felder des QS-Filter-Eingangsdatensatzes nach §301

| Feld | Beschreibung | M/K | Zeichenlänge | BasisTyp |
|-------------------------------|---------------------------------------------------------------------------------------------------------------------------------------|-----|--------------|----------------------------|
| Behandlungsfall (FALL) | | | | |
| FALLNUMMER | Fallnummer | M | 15 | TEXT |
| FALLART | Fallart 1: ENTGELTARTAMB <> LEER UND ENTGELTARTSTAT = LEER 2: ENTGELTARTSTAT <> LEER UND ENTGELTARTAMB = LEER | M | 1 | NUMSCHLUESSEL (Fallart) |
| AUFNGRUND | Aufnahmegrund gem. §301 SGB V | K | 2 | SCHLUESSEL (AufnGrund) |
| AUFNDATUM | Datum der Aufnahme Krankenhaus | K | 10 | DATUM |
| ENTLDATEM | Datum der Entlassung Krankenhaus | K | 10 | DATUM |
| PATALTER | Alter in Jahren am Auf- nahmetag | 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 Auf- nahmetag | K | 5 | GANZEZAHL |
| GEWICHT | Aufnahmegewicht in Gramm | K | 6 | GANZEZAHL |
| DAUBEAT | Beatmungstunden | K | 4 | GANZEZAHL |

¹¹ Vereinbarung über die Übermittlung von DRG-Daten nach §21 KHEntgG. Stand: 30. November 2012. Siegburg: Institut für das Entgeltsystem im Krankenhaus (InEK gGmbH). Online verfügbar unter http://www.gdrg.de/cms/Datenlieferung_gem._21_KHEntgG/Dokumente_zur_Datenlieferung/Datensatzbeschreibung.

| Feld | Beschreibung | M/K | Zeichenlänge | BasisTyp |
|-----------------------------------------------------|----------------------------------------------------------------|-----|--------------|---------------------------------------|
| ENTLZEIT | Entlassungs-/ Verlegungsuhrzeit | K | 5 | UHRZEIT |
| STANDORT | entlassender Standort | M | 2 | TEXT |
| KASSEIKNR | Institutionskennzeichen der Krankenkasse der Versichertenkarte | K | 9 | GANZEZAHL |
| Prozedurangaben (wiederholbar) (PROZ) | | | | |
| FALLNUMMER | Fallnummer | M | 15 | TEXT |
| OPDATUM | Datum der Operation | M | 10 | DATUM |
| OPS | Prozedurschlüssel (OPS-Kode) | M | 13 | SCHLUESSEL (OPS) |
| Diagnoseangaben (wiederholbar) (DIAG) | | | | |
| FALLNUMMER | Fallnummer | M | 15 | TEXT |
| ICD | Diagnoseschlüssel (ICD-Kode) | M | 9 | SCHLUESSEL (ICD) |
| DIAGART | Diagnoseart | M | 2 | SCHLUESSEL (DiagArt) |
| SEKDIAG | Sekundär-Diagnoseschlüssel (Sekundär-Kode) | K | 9 | SCHLUESSEL (ICD) |
| Entgeltangaben (wiederholbar) (ENTGELT) | | | | |
| FALLNUMMER | Fallnummer | M | 15 | TEXT |
| ENTGELTARTS-TAT | Entgeltschlüssel stationär gem. § 301 SGB V (1.-3. Stelle) | K | 3 | NUMSCHLUESSEL (EntgeltSchlüssel-Stat) |
| ENTGELT-ARTAMB | Entgeltschlüssel ambulant gem. § 301 SGB V | K | 8 | NUMSCHLUESSEL (EntgeltSchlüssel-Amb) |
| EBMDATUM | Leistungstag (Entgelt) | K | 10 | DATUM |
| Fachabteilungsangaben (wiederholbar) (ABTLG) | | | | |
| FALLNUMMER | Fallnummer | M | 15 | TEXT |
| FACHABT | Fachabteilung | M | 4 | SCHLUESSEL (Fachabt) |

| Feld | Beschreibung | M/K | Zeichenlänge | BasisTyp |
|--------------|--------------------------------------------------------|-----|--------------|----------|
| FABAUFNDATUM | Datum der Aufnahme in die Fachabteilung | M | 10 | DATUM |
| FABAUFNZEIT | Uhrzeit der Aufnahme in die Fachabteilung | M | 5 | UHRZEIT |
| FABENTLDATUM | 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 sind¹².

Für den externen Schlüssel EntgeltSchlüsselStat sind dreistellige Schlüsselwerte hinterlegt. Diese dienen im Rahmen der Spezifikation der Identifikation belegärztlicher Fälle.

Primär und Sekundär Diagnosen

Seit Einführung des Verfahrens QS-WI in der Spezifikation 2017 ist das Feld SEKDIAG (bzw. die Syntaxvariablen SEKDIAG und HSEKDIAG) im Eingangsdatensatz enthalten. Für die jeweiligen Module sind Einschlussdiagnosen nach Primär- und Sekundär Diagnosen im Sinne der Kreuz-Stern-Notation zu differenzieren. Die Zusammengehörigkeit bestimmter Sekundär Diagnosen zu bestimmten Primär Diagnosen ist hierbei zu berücksichtigen. Die in den Auslösekriterien referenzierten Felder DIAG bzw. HDIAG beinhalten weiterhin sowohl Primär- als auch Sekundär Diagnosen

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

Der QS-Filter-Eingangsdatensatz nach §295 SGB V bezieht sich auf Fälle, die kollektivvertraglich durch einen niedergelassenen Leistungserbringer (Arztpraxis, MVZ, auch Eingriffe durch Belegärzte oder durch Krankenhausärzte im Rahmen einer KV-Ermächtigung)¹³ 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 können¹⁴. Einzige Ausnahme sind die Felder PATALTER (lässt sich aus Inhalten des KVDT-Datensatzes berechnen) und BEHANDLFALLID (kann zur

¹² Aktuelle Version abrufbar im Download-Bereich der DKG-Homepage (<http://www.dkgev.de>)

¹³ Ambulante Leistungen nach §295 SGB V, §116 SGB V, stationäre Fälle nach §121 Abs. 3 SGB.

¹⁴ 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).

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 vier 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 9: Felder des QS-Filter-Eingangsdatensatzes nach § 295

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

| Feld | Beschreibung | M/K | Zeichenlänge | BasisTyp |
|-------------------------------------|----------------------------------|-----|--------------|----------------------------------------|
| 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.

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 leistungsbereichsbezogenen 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 Einrichtung nur einmal ausgelöst wird. Es können aber mehrere verschiedenartige Module ausgelöst werden. Ein Krankenhaus kann beispielsweise als stationäre Einrichtung für das Modul NWIES und als ambulant operierende Einrichtung für das Modul NWIEA dokumentationspflichtig werden.

Dokumentationspflicht einer Einrichtung

Eine Einrichtung ist dokumentationspflichtig, wenn für sie mindestens ein QS-Modul dokumentationspflichtig ist. Ein QS-Modul wird als dokumentationspflichtig identifiziert, wenn mindestens ein Fall den definierten Auslösekriterien entspricht.

Dokumentationspflicht eines Moduls

Ein Modul (Datensatz) ist dann dokumentationspflichtig, wenn folgende Bedingungen zutreffen:

- Bei der Fehlerprüfung tritt kein Fehler auf (Abschnitt B 1.4.5).
- 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 beschriebenen Stufen der Dokumentationsverpflichtung.

1.4.2 Leistungsbereichsbezogene Einschlusskriterien

Überblick:

Für jede QS-Datensatz-Definition (= Modul) sind in der Tabelle `ModulAusloeser` 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.



Achtung

Einschlusskriterien mit KASSEIKNR

Leistungsbereichsbezogene Einschlusskriterien können das Feld `KASSEIKNR` enthalten, mithilfe dessen geprüft wird, ob es sich um einen GKV-Patienten handelt. Liegt die `KASSEIKNR` bzw. die elektronische Gesundheitskarte (eGK) bei der Prüfung der definierten Kriterien noch nicht vor, erlischt **nicht** die Dokumentationspflicht. Die Auslösebedingung ist bei Vorliegen der `KASSEIKNR`/eGK erneut zu prüfen.

Üblicherweise entspricht ein QS-Datensatz genau einem QS-Filter-Leistungsbereich. Zurzeit gibt es folgende Ausnahmen:

Ausnahmen für einzelne QS-Filter-Leistungsbereiche:

- Für die QS-Filter-Leistungsbereiche `NWIEA_LKG`, `NWIEA_KV`, `NWIES_LKG` und `NWIES_KV` sind keine QS-Datensätze definiert. Es handelt sich um QS-Filter-Leistungsbereiche, die Untermengen der Dokumentationsmodule `NWIEA` und `NWIES` nach der Art der Abrechnung darstellen.

Definition in der Datenbank

Tabelle 10 gibt einen Überblick über die Tabelle `ModulAusloeser` der Spezifikationsdatenbank.

Tabelle 10: Struktur der Tabelle ModulAusloeser

| Feldname | Datentyp | Bemerkung |
|-----------------------|----------|-------------------------------------------------------------------------------|
| idModulAusloeser | INTEGER | Primärschlüssel |
| name | TEXT | Technischer Name des QS-Filter-Leistungsbe- reichs |
| bezeichnung | TEXT | Bezeichnungstext der Auslösebedingung |
| bedingung | MEMO | Auslösebedingung für den QS-Filter-Leistungs- bereich |
| textDefinition | MEMO | Medizinisch-inhaltliche Definition bzw. Erläute- rung der Auslösebedingung |
| fkModul | INTEGER | Bezug zum Modul |
| verpflichtend | BOOLEAN | Wenn WAHR, ist das betreffende Modul bun- desweit verpflichtend |
| fkAdminKriterium | INTEGER | Bezug zu den administrativen Abgrenzungskri- terien |
| fkDs | INTEGER | Bezug zum entsprechenden Sollstatistikdaten- satz |
| internBild | TEXT | Bild zum Modul |
| ambulant | BOOLEAN | Ist die Fallart ambulant? |
| stationaer | BOOLEAN | Ist die Fallart stationär? |
| krankenhausabrechnung | BOOLEAN | Wird der Fall stationär abgerechnet? |
| selektiv | BOOLEAN | Wird der Fall ambulant selektivvertraglich abge- rechnet? |
| kollektiv | BOOLEAN | Wird der Fall ambulant kollektivvertraglich ab- gerechnet? |

Identifizierung der Fallart und der Art der Leistungserbringung

Zur Abbildung sektorenübergreifender Module wurden die booleschen Attribute zur Fallart (*ambulant*, *stationaer*) und zur Art der Leistungserbringung (*krankenhausabrechnung*, *kollektiv*, *selektiv*) in die Tabelle *ModulAusloeser* integriert. Diese können wie folgt identifiziert werden:

ambulant

Dieses Attribut ist WAHR, wenn Feld FALLART = 1.

stationaer

Dieses Attribut ist WAHR, wenn Feld FALLART = 2.

krankenhausabrechnung

Dieses Attribut ist WAHR, wenn Feld ENTGELTARTAMB <> LEER ODER (ENTGELTARTSTAT <> LEER UND ENTGELTARTSTAT KEINSIN STAT_BELEG_ENTGELT).

kollektiv/selektiv¹⁵

Diese Attribute können über das Feld EBM identifiziert werden. Das Attribut selektiv/kollektiv ist WAHR, wenn EBM <> LEER. In diesem Fall ist das Attribut krankenhausabrechnung = FALSCH.

1.4.3 Administrative Einschlusskriterien**Überblick**

Die administrativen Regelungen beschreiben diejenigen Auslösebedingungen, welche über die modulspezifischen ICD/OPS-Kriterien (Abschnitt B 1.4.2) hinausgehen. Moduldokumentationen werden für eine Einrichtung (Prüfung auf Fallebene) nur dann ausgelöst, wenn

1. der Patient stationär/ambulant aufgenommen/behandelt wird,
2. der Behandlungsfall einen bestimmten zeitlichen Rahmen aufweist,
3. der Behandlungsfall keine Verbringungsleistung für Dritte ist.

Zu 1.

Stationär aufgenommen: Die Bedingung wird aus der Angabe im Datenfeld Aufnahmegrund (Abschnitt B 1.3.1) abgeleitet:

```
AUFNGRUND NICHTIN ( '03'; '04' ) UND AUFNGRUND <> LEER
```

Diese Teilbedingung ist Teil der administrativen Kriterien (siehe Attribut bedingung in Tabelle 11). Der zeitliche Rahmen wird bei stationären Fällen über das Aufnahme- und das Entlassungsdatum definiert.

Zu 2.¹⁶**Beispiel:**

Stationäre Patienten, welche im 1. oder 2. Quartal 2017 des Jahres 2017 aufgenommen und entlassen sind:

```
AUFNDATUM >= '01.01.2017' UND AUFNDATUM <= '31.06.2017' UND
ENTLDDATUM >= '01.01.2017' UND ENTLDDATUM <= '31.06.2017'
```

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.

¹⁵ Die Attribute kollektiv und selektiv sollen die Unterscheidung beider Attribute ermöglichen. Die Attribute können ggf. nicht durch jeden Anbieter differenziert werden. Eine spätere Konkretisierung wird in der Spezifikation vorgenommen, sobald eindeutige Marker existieren bzw. identifiziert werden konnten, die eine bundesweite Differenzierung zwischen selektivvertraglichen und kollektivvertraglichen Fällen ermöglichen.

¹⁶ Verbindlich sind die in der Spezifikationsdatenbank hinterlegten Kriterien.

Definition in der Datenbank

Tabelle 11 gibt einen Überblick über wichtige Attribute der Tabelle `AdminKriterium`, welche die administrativen Abgrenzungskriterien definiert.

Tabelle 11: Struktur der Tabelle `AdminKriterium`

| Feldname | Datentyp | Bemerkung |
|-------------------------------|----------|-----------------------------------------------------|
| <code>idAdminKriterium</code> | INTEGER | Primärschlüssel |
| <code>name</code> | TEXT | Technischer Name des administrativen Kriteriums |
| <code>bedingung</code> | MEMO | Auslösebedingung für den QS-Filter-Leistungsbereich |
| <code>info</code> | TEXT | Bezeichnungstext des administrativen Kriteriums |

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 13).

Typen

Jede Variable hat einen Basistyp. Tabelle 12 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 12: Basistypen der Variablen

| Basistyp | Bezeichnung | Beispiele (Literele) |
|-----------|---------------------------------------|-------------------------------|
| BOOL | Boolesche Variable | WAHR, FALSCH |
| TEXT | Zeichenkette (String) | „Spezifikation“ |
| GANZEZAHL | ... -2, -1, 0, 1, 2, 3, ... | 1 |
| ZAHL | Zahl (mit oder ohne Nachkommastellen) | Erfassung: 25, 4 oder -100, 8 |

| Basistyp | Bezeichnung | Beispiele (Literale) |
|---------------|-----------------------------------------------|-----------------------------|
| | | 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' |

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- und Abrechnungslisten

Fest definierte Listen von Diagnosen, Prozeduren 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 `OPListe` (Attribut `name`) und von Abrechnungslisten in der Tabelle `EBMListe` (Attribut `name`) definiert.

Die jeweiligen Codes (nach ICD-10-GM, OPS bzw. EBM) sind in der durch Fremdschlüsselbeziehungen verknüpften Tabelle `ICDWert`, `OPSWert` 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 13 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 13: Präzedenz und Assoziativität der Operatoren¹⁷

| 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)¹⁸.

Beispiel:

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

```
PROZ EINSIN TON OPS
```

String-Vergleich bei Diagnosen, Prozeduren und Abrechnungsziffern

Die Operation EINSIN basiert für Diagnosen, Prozeduren oder Abrechnungsziffern auf einem Vergleich von Zeichenketten (String-Vergleich): Für jedes Element der linken Liste wird über einen String-Vergleich geprüft, ob es in der rechten Liste enthalten ist.

¹⁷ 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.

¹⁸ Dieser Operator ist redundant, da er auch durch Negation des EINSIN-Operators abgedeckt 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 Codes enthalten. Entsprechend muss sichergestellt sein, dass auch die Codes aus dem Eingangsdatensatz terminal sind. Darauf ist besonders zu achten, wenn Daten ungeprüft aus Subsystemen übernommen wurden.

Beispiel:

Folgende Operationen führen zum gleichen Ergebnis

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

Eine analoge Regelung gilt für die Seitenlokalisationen der OPS-Kodes.

1.4.5 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 14: Fehlercodes des QS-Filters

| Beschreibung des Fehlers | Fehlercode | Formale Fehlerbedingung | Fehlermeldung |
|-------------------------------------------------------------------------|------------|-------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| Überprüfung des Formats | 1 | Abschnitt B 2.3.8 | Der Wert '<WERT>' des Datenfeldes <Feld.name> '<TdsFeld.bezeichnung>' ist kein gültiger <BasisTyp.name>-Wert (<BasisTyp.bezeichnung> <BasisTyp.format>). |
| Überprüfung der Feldlänge | 2 | Abschnitt B 2.3.8 | Der Wert '<WERT>' des Datenfeldes <Feld.name> '<TdsFeld.bezeichnung>' überschreitet die zulässige Feldlänge <Feld.laenge>. |
| Sind in den Datenfeldern mit internen und externen Schlüsseln (BasisTyp | 3 | Abschnitt B 2.3.8 | Ungültiger Schlüsselcode <Wert> des Schlüssels <Schlüssel.name> im Datenfeld <Feld.name> '<TdsFeld.bezeichnung>' ! |

| Beschreibung des Fehlers | Fehler-kode | Formale Fehlerbedingung | Fehlermeldung |
|------------------------------------------------------------------------------------|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SCHLUESSEL oder NUM-SCHLUESSEL) gültige Schlüsselkodes eingetragen? ¹⁹ | | | |
| Überprüfung numerischer Wertebereiche | 4 | Abschnitt B 2.3.8 | Der Wert '<WERT>' des Datenfeldes <Feld.name> '<TdsFeld.bezeichnung>' ist kleiner als '<Feld.min>' oder Der Wert '<WERT>' des Datenfeldes <Feld.name> '<TdsFeld.bezeichnung>' ist größer als '<Feld.max>' |
| Ist ein Muss-Feld ausgefüllt? | 5 | Abschnitt B 2.3.8 : Attribut <TdsFeld.fkMussKann> | Das Datenfeld <Feld.name> '<TdsFeld.bezeichnung>' muss einen gültigen Wert enthalten. |
| Ist der Fall dem Erfassungsjahr der Spezifikation zugeordnet? (Spezifikation 2017) | 6 | MODUL = 'NWIES' UND FALLART = 2) UND AUFNDATUM <'01.01.2017' ODER AUFNDATUM> '31.06.2017') ODER (MODUL = 'NWIEA' UND FALLART <> 2 UND (OPDATUM < '01.01.2017' ODER OPDATUM > '31.06.2017')) | Der Fall ist im Jahr 2017 nicht dokumentationspflichtig: Aufnahmedatum/OP-Datum = <Wert> |

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.

¹⁹ 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 sichergestellt sein.

Es wird beim Auftreten eines Fehlers nicht weiter geprüft, ob für den Behandlungsfall Moduldokumentationen ausgelöst werden.

2 QS-Dokumentation

Die vorliegenden Spezifikationskomponenten für die QS-Dokumentation definieren die Vorgaben zur Datenerfassung, Plausibilitätsprüfung und Datenübermittlung für die externe vergleichende Qualitätssicherung im Rahmen der Qesü-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 und die Dokumentationsbögen. Die Delta-Tabellen in der Datenbank zur QS-Dokumentation stellen geänderte, gelöschte und neue Datenbankinhalte im Vergleich zur letzten gültigen Version des Vorjahres sowie, sofern vorhanden, zur letzten Version des aktuellen Erfassungsjahres dar.

2.1 Anmerkungen zur Struktur der Spezifikation zur QS-Dokumentation

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

`<Erfassungsjahr>_Qesue_QSDOK_V<Versionsnummer>.mdb`

`<Erfassungsjahr>` bezeichnet das Jahr, in dem die QS-Dokumentation stattfindet. `<Versionsnummer>` bezeichnet die 2-stellige Versionsnummer (z.B. 01).

Beispiel:

Im Erfassungsjahr 2017 ist die Spezifikation `2017_Qesue_QSDOK_V01.mdb`²⁰ gültig.

Weiterführende Erläuterungen zum Benennungsschema für Spezifikationskomponenten sind der Einleitung, Abschnitt 1.1.2, zu entnehmen. Die Tabellenstruktur der Spezifikationsdatenbank wird in Abschnitt B Komponenten beschrieben.

Folgende Attribute treten in vielen Tabellen auf:

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

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 2), die in den nachfolgenden Abschnitten erläutert werden.

²⁰ Die Versionsnummer der gültigen Spezifikation (z.B. V01, V02 usw.) ist dem zuletzt veröffentlichten Update zu entnehmen.

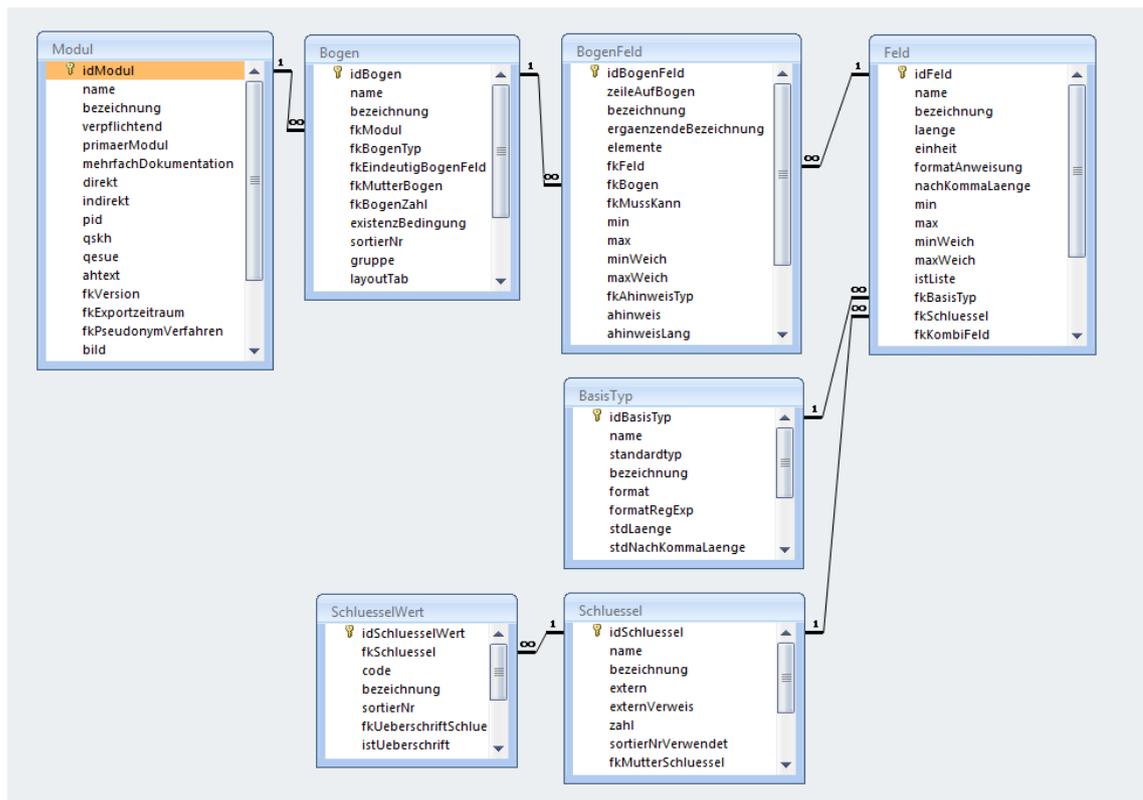


Abbildung 2: Tabellen und Relationen der Datenfeldbeschreibung

Die Abfragen Datenfeldbeschreibung und DatenfeldbeschreibungFürEinModul der Access-Datenbank ermöglichen den Überblick über diese Struktur.

Die Beschreibung der Datenfelder hat folgende Ziele:

- Bereitstellung der Informationen, welche für die Programmierung des Eingabeformulars und für die Sicherung der eingegebenen Daten nötig sind
- Vermeidung von Redundanzen
- Typisierung der Felder nach fachlichen und datentechnischen Kriterien

Das für den Anwender wichtigste Merkmal ist die Bezeichnung des Datenfelds (Attribut BogenFeld.bezeichnung).

Die Datenfeldbeschreibung ist teilweise auf dem jeweiligen Dokumentationsbogen eines Moduls („Bogensicht“) abgebildet. Zu beachten ist dabei, dass die „Bogensicht“ lediglich die Sicht der medizinischen Fachgruppen, die die Module entwickeln, darstellt. Bei verteilten Softwarelösungen²¹ 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.

²¹ Es besteht für Kassenärztliche Vereinigungen die Möglichkeit den Vertragsärzten in ihrem Zuständigkeitsbereich Anwendungen zur Erfassung und Übertragung von Daten der einrichtungsbezogenen QS-Dokumentation, z.B. webbasiert, zur Verfügung zu stellen. Im Krankenhaussektor wird von Softwareherstellern eine QS-Filtersoftware umgesetzt.

**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 Prozessablauf²² eines Leistungserbringers einordnet.

2.2.1 Dokumentationsmodule (Datensätze)

Ein Modul der Spezifikation enthält die Datensatzdefinition von einem oder mehreren Verfahren (Qesü-RL). Abhängig von (inhaltlich oder organisatorisch) abzugrenzenden Bereichen kann ein Verfahren mehrere Module umfassen. Für eine Einrichtung kann eine oder mehrere Moduldokumentationen angelegt werden, 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 15: Struktur der Tabelle *Modul*

| Feldname | Datentyp | Bemerkung |
|-----------------------|--------------|------------------------------------------------------------------------------------------------------|
| idModul | INTEGER | Primärschlüssel |
| Name | TEXT (32) | Eindeutiger technischer Name |
| bezeichnung | TEXT (255) | Erläuternde Bezeichnung |
| verpflichtend | BOOLEAN | Besteht für das Modul eine QS-Dokumentationsverpflichtung? |
| primaerModul | BOOLEAN | Ist das Modul ein Primärmodul? |
| mehrfachDokumentation | BOOLEAN | Ist ein mehrfaches Anlegen eines gleichartigen Datensatzes pro Einrichtung zulässig (WAHR / FALSCH)? |
| direkt | BOOLEAN | Handelt es sich um ein direktes Datenexportverfahren? |
| indirekt | BOOLEAN | Handelt es sich um ein indirektes Datenexportverfahren? |
| Pid | BOOLEAN | Handelt es sich um ein Modul zur Follow-up-Erhebung? ²³ |

²² Zum Beispiel in den Prozessablauf eines Krankenhauses.

²³ Gemeint ist eine Follow-up-Erhebung mittels PID-Verfahren.

| Feldname | Datentyp | Bemerkung |
|----------------------|-------------|-----------------------------------------------------------------|
| Qskh | BOOLEAN | Handelt es sich um ein Modul der QSKH-Richtlinie? |
| qesue | BOOLEAN | Handelt es sich um ein Modul der Qesü-Richtlinie? |
| ahtext | MEMO | Einleitender Text für den Ausfüllhinweis eines Moduls |
| fkVersion | INTEGER | Gültige Version des jeweiligen Moduls |
| fkExportzeitraum | INTEGER | Exportzeitraum (Exportzeit- raum . exportBis) eines Moduls |
| fkPseudonymVerfahren | INTEGER | Pseudonymisierung von FU-Verfahren ²⁴ |
| Bild | TEXT (20) | Modulspezifisches Bild |

In der Tabelle Modul wird der Exportzeitraum für ein Modul angegeben (Attribut Modul . fkExportzeitraum). Die hinterlegten Zeitpunkte sind in der Tabelle Exportzeitraum (Attribut exportBis) definiert.

Auslösung der Moduldokumentation

Der auslösende Sachverhalt für die Dokumentationspflicht ist in der Spezifikation für den QS-Filter definiert. Dieser greift zu diesem Zweck unter anderem auf ICD- und OPS-Kodes (z.B. Haupt- und Nebendiagnosen und Prozeduren) und Gebührenordnungspositionen (EBM-Kodes) zurück, die z.B. im Krankenhausinformationssystem (KIS) oder in den Abrechnungsdaten der Kassenärztlichen Vereinigungen verfügbar sind. Die Dokumentationspflicht der Module des Verfahrens QS-WI wird unter anderem über Einschlussdiagnosen ermittelt, die nach Primär- und Sekundärdiagnosen im Sinne der Kreuz-Stern-Notation zu differenzieren sind. Die Zusammengehörigkeit bestimmter Sekundärdiagnosen zu bestimmten Primärdiagnosen ist bei der Auslösung zu berücksichtigen.

Primärmodule

Für Primärmodule sind in der Spezifikation für QS-Filter-Software Auslösebedingungen definiert.

2.2.2 Teildatensätze

Die Begriffe „Teildatensatz“ und „Bogen“ werden synonym gebraucht. In den der Illustration dienenden Dokumentationsbögen werden alle Teildatensätze aufgeführt.

Benennung von Teildatensätzen

Ein Teildatensatz wird durch die folgende Kombination von Modulnamen und Bogennamen identifiziert und angesprochen:

<Modul . name> : <Bogen . name>

²⁴ Gemeint ist ein Follow-up-Verfahren anhand von PID-Daten.

Beispiele:

NWIEA : Bist der Basisbogen des Moduls Vermeidung nosokomialer Infektionen:
Postoperative Wundinfektionen (einrichtungsbezogen ambulant)

Bogentyp

Der Kerndatensatz besteht aus mindestens einem Basisteildatensatz und kann durch einen oder mehrere Teildatensätze ergänzt werden. Das Attribut `Bogen.fkBogenTyp` definiert für jeden Teildatensatz seine Rolle im und seine Zugehörigkeit zum Kerndatensatz. In Tabelle 16 sind die Bezeichnungen der einzelnen Bogentypen definiert.

Tabelle 16: Inhalte der Tabelle `BogenTyp`

| <code>idBogenTyp</code> | Bezeichnung |
|-------------------------|--------------------------------------------------------------------------|
| B | Basisteildatensatz (Teil des Kerndatensatzes) |
| K | Teildatensatz ist Teil des Kerndatensatzes und kein Basisteildatensatzes |
| O | Teildatensatz ist Teil des optionalen Datensatzes |

Die Module NWIES und NWIEA verfügen jeweils nur über einen Basisteildatensatz.

Weiterführende Erläuterungen zu Teildatensätzen sind in der Technischen Dokumentation zur Basisspezifikation wird Leistungserbringer zu entnehmen.

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 Bezeichnung²⁵ 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 `Field` aufgelisteten Felder inhaltlich gleiche Dokumentationsfelder mehrerer Module (Abschnitt B 2.2.1), der datentechnische Typ (`BasisTyp`) charakterisiert das Format des Feldes (Abschnitt B 2.2.1).

Jedes Datenfeld hat zwingend einen Bezug zu einem Teildatensatz und zu einem technischen Feld. Weitere Eigenschaften sind die Bogenfeldbezeichnung und die fortlaufende Nummer im Teildatensatz. Die Datenfelder sind in der Tabelle `BogenField` gespeichert.

Identifizierendes Merkmal eines Datenfelds ist eine Kombination aus `fkBogen` und `fkField`. Das bedeutet, dass das Datenbankschema gewährleistet, dass der technische Feldname (`Field.name`) in einem Teildatensatz maximal einmal vorkommt. Per definitionem muss ein

²⁵ Gegebenenfalls im Kontext der Überschriften (Abschnitt B 1.3.4).

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

Tabelle 17: Struktur der Tabelle `BogenFeld`

| Feldname | Datentyp | Bemerkung |
|-------------------------------------|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>idBogenFeld</code> | INTEGER | Primärschlüssel |
| <code>zeileAufBogen</code> | DOUBLE | Zeile in dem Dokumentationsbogen. |
| <code>bezeichnung</code> | TEXT | Beschreibender Text zum Feld auf dem Dokumentationsbogen. Wenn der Feldinhalt leer ist, so wird der Inhalt des gleichnamigen Felds in der Tabelle <code>Feld</code> genommen. |
| <code>ergaenzendeBezeichnung</code> | TEXT | Optionale ergänzende Bezeichnung zu einem Bogenfeld. |
| <code>elemente</code> | INTEGER | Anzahl der Elemente bei Listefeldern |
| <code>fkFeld</code> | INTEGER | Fremdschlüssel zu dem Teildatensatz und zu dem Feld, bilden zusammen die identifizierenden Merkmale |
| <code>fkBogen</code> | INTEGER | |
| <code>fkMussKann</code> | TEXT (1) | M oder K, Unterscheidung zwischen Muss- und Kann-Feldern |
| <code>min</code> | DOUBLE | Harte Untergrenze des Wertebereichs eines numerischen Datenfeldes (modulspezifisch). Die Definition ist optional. |
| <code>max</code> | DOUBLE | Harte Obergrenze des Wertebereichs eines numerischen Datenfeldes (modulspezifisch). Die Definition ist optional. |
| <code>minWeich</code> | DOUBLE | Weiche Untergrenze des Wertebereichs eines numerischen Datenfeldes (modulspezifisch). Die Definition ist optional. |
| <code>maxWeich</code> | DOUBLE | Weiche Obergrenze des Wertebereichs eines numerischen Datenfeldes (modulspezifisch). Die Definition ist optional. |
| <code>ahinweis</code> | TEXT (32) | Name des HTML-Ausfüllhinweises ohne Endung <code>.htm</code> (Abschnitt B 2.2.5) |

Muss- und Kann-Felder

Jedes Bogenfeld ist als Muss- oder Kann-Feld zu deklarieren:

- Ein Muss-Feld (M) muss innerhalb eines angelegten Teildatensatzes immer ausgefüllt sein (Abschnitt B 2.2.2).²⁶
- Kann-Felder (K) sind optionale Felder.

²⁶ In jedem Muss-Feld muss für jeden angelegten Teildatensatz einmal eine Angabe erfolgen.

- Abhängige Muss-Felder (\mathcal{K}) müssen nur unter bestimmten Bedingungen ausgefüllt werden. Wenn also logische Sachverhalte dem Ausfüllen von Kann-Feldern entgegenstehen, so dürfen sie nicht ausgefüllt werden. Diese Felder unterliegen Feldgruppenregeln und verfügen wie optionale Felder über den Attributwert \mathcal{K} .

Anzahl der Elemente von Listenfeldern

Das Attribut `elemente` ist nur relevant bei von Listenfeldern (vgl. Attribut `istListe` der Tabelle `Feld`) abgeleiteten Bogenfeldern (Bogenfeldlisten). Es gibt die Größe der Bogenfeldliste an. Wenn für eine Bogenfeldliste das Attribut `elemente` leer ist, so ist die Größe per Definition 1.

Wenn ein Listenfeld als Muss-Feld deklariert ist, so ist nur das erste Exportfeld der Liste ein Muss-Feld, die restlichen Elemente sind Kann-Felder. Wenn ein Listenfeld als Kann-Feld deklariert ist, so sind alle weiteren exportierten Elemente ebenfalls Kann-Felder.

Felder – ein erster Schritt zur Prozess- und Datenintegration

Die Tabelle `Feld` (Tabelle 18) 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`²⁷ beschrieben.

Identifizierendes Attribut eines Felds ist allein sein technischer Name (Attribut `name`). Dies ist wichtig für die Eindeutigkeit von Feldnamen innerhalb eines Moduls: Felder mit unterschiedlichen Typen oder unterschiedlichen Schlüsseln müssen unterschiedliche Namen haben.

Ein Feld kann als Skalar oder als Liste definiert sein. Diese Eigenschaft wird über das Attribut `istListe` gesteuert. Jedes von einem Listenfeld abgeleitete Bogenfeld ist automatisch eine Liste.²⁸ Die Anzahl der Elemente des von einem Feld abgeleiteten Bogenfelds wird über das Attribut `elemente` der Tabelle `BogenFeld` gesteuert.

Insbesondere für die Verwendung der richtigen Operatoren in den Plausibilitätsregeln und Feldgruppen ist die Listendefinition eines Felds wichtig.

²⁷ Das Attribut `bezeichnung` ist ein Standardtext für das gleichnamige Attribut der Tabelle `BogenFeld`. Im Eingabeformular wird die Bezeichnung aus der Tabelle `BogenFeld` angezeigt.

²⁸ Man beachte die Besonderheiten der Listenfelder beim Datenexport und in der Syntax der Plausibilitätsregeln.

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 18: Struktur der Tabelle `Feld`

| Feldname | Datentyp | Bemerkung |
|------------------------------|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>idFeld</code> | INTEGER | Primärschlüssel |
| <code>name</code> | TEXT | Technischer Name |
| <code>bezeichnung</code> | 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>) |
| <code>laenge</code> | INTEGER | Anzahl der Zeichen in der Feldeingabemaske, enthält beim Typ <code>ZAHL</code> auch das Komma, bei <code>SCHLUESSEL</code> die Trennzeichen |
| <code>einheit</code> | TEXT (50) | Einheit des Felds (z.B. mm, Stunden) |
| <code>formatAnweisung</code> | TEXT | Regulärer Ausdruck für die Formatprüfung (z.B. <code>[0-9]{9}</code>) |
| <code>nachKommaLaenge</code> | INTEGER | Anzahl der Nachkommastellen in der Feldeingabemaske (muss kleiner als <code>laenge</code> sein) |
| <code>min</code> | DOUBLE | Harte Untergrenze des Wertebereichs eines numerischen Datenfelds (modulübergreifend). Die Definition ist optional. |
| <code>max</code> | DOUBLE | Harte Obergrenze des Wertebereichs eines numerischen Datenfelds (modulübergreifend). Die Definition ist optional. |
| <code>minWeich</code> | DOUBLE | Weiche Untergrenze des Wertebereichs eines numerischen Datenfelds (modulübergreifend). Die Definition ist optional. |
| <code>maxWeich</code> | DOUBLE | Weiche Obergrenze des Wertebereichs eines numerischen Datenfelds (modulübergreifend). Die Definition ist optional. |
| <code>istListe</code> | BOOLEAN | Wenn <code>istListe = WAHR</code> , so sind die vom betreffenden Feld abgeleiteten Bogenfelder Listenfelder. |
| <code>fkBasisTyp</code> | INTEGER | Fremdschlüssel zur Tabelle <code>Basistypen</code> |
| <code>fkSchluessel</code> | INTEGER | Fremdschlüssel zur Tabelle <code>Schlüsseltypen</code> |

| Feldname | Datentyp | Bemerkung |
|-------------|----------|-----------------------------------------------------------------------------------------|
| fkKombiFeld | INTEGER | Optionaler Fremdschlüssel auf ein anderes Feld, welches Kombinationsfelder kennzeichnet |

Kombinationsfelder

Für manche Bogenfelder ist zwingend vorgeschrieben, dass sie innerhalb eines Moduls in Kombination mit einem anderen Bogenfeld existieren. Die Definition von Kombinationsfeldern geschieht mithilfe des optionalen Fremdschlüssels `fkKombiFeld` in der Tabelle `Feld`.

Basistypen

Das Hauptmerkmal eines Basistyps ist der technische Typ eines Eingabefelds (z.B. Zeichenkette, numerischer Typ, Datum usw.). Wichtiges Charakteristikum ist die Beschreibung des Eingabeformats. Die Basistypen sind Voraussetzung für die Beschreibung einer formalen Regelsyntax (Abschnitt B 2.3.2).

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

Tabelle 19: Struktur der Tabelle `Basistyp`

| Feldname | Datentyp | Bemerkung |
|--------------------|----------|-----------------------------------------------------------------------------------------------------------------|
| idBasistyp | INTEGER | Primärschlüssel |
| name | TEXT | Technischer Name (muss eindeutig sein) |
| standardtyp | TEXT | Entsprechender Standarddatentyp |
| bezeichnung | TEXT | Beschreibender Text |
| format | TEXT | Formatdefinition, z.B. TT.MM.JJJJ beim Basistyp <code>Datum</code> |
| formatRegExp | TEXT | Regulärer Ausdruck für die Formatprüfung |
| stdLaenge | INTEGER | Vorschlagsfeld für das gleichnamige Feld in der Tabelle <code>Feld</code> (einschließlich Vorzeichen und Komma) |
| stdNachKommaLaenge | INTEGER | Vorschlagsfeld für das gleichnamige Feld in der Tabelle <code>Feld</code> |



Hinweis

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

Schlüssel

Identifizierendes Merkmal eines Schlüssels (Kodesystem) ist sein technischer Name. Die meisten Schlüsselcodes sind in der Tabelle `Schluesse1` definiert.

Tabelle 20: Struktur der Tabelle `Schluesse1`

| Feldname | Datentyp | Bemerkung |
|---------------------------------|----------|--------------------------------------------------------------------------------------------------------------------------------------------|
| <code>idSchluesse1</code> | INTEGER | Primärschlüssel |
| <code>name</code> | TEXT | Technischer Name (muss eindeutig sein) |
| <code>bezeichnung</code> | TEXT | Beschreibender Text |
| <code>extern</code> | BOOLEAN | Zeigt an, ob der Schlüssel in der Tabelle <code>Schluesse1</code> (= FALSCH) oder in einer externen Tabelle gespeichert (= WAHR) ist. |
| <code>externVerweis</code> | TEXT | Verweis auf die Quelle des externen Schlüssels |
| <code>zahl</code> | BOOLEAN | Wenn WAHR, sind die Werte im Attribut <code>code</code> der zugehörigen Schlüsselwerte als ganze Zahl kodiert, ansonsten als Zeichenkette. |
| <code>sortierNrVerwendet</code> | BOOLEAN | Flag, das anzeigt, ob für die Reihenfolge das Attribut <code>sortierNr</code> der Tabelle <code>Schluesse1</code> -Wert herangezogen wird. |
| <code>fkMutterSchluesse1</code> | INTEGER | Referenz auf einen übergeordneten Schlüssel. |

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

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.dimdi.de). Diese externen Schlüsselkataloge werden nicht vom IQTIG bereitgestellt und somit auch nicht verantwortet.

Hinweise zu den Bezugsquellen sind in der Spalte `externVerweis` zu finden. Ein Verweis auf eine Bezugsquelle kann unabhängig vom Attribut `extern` angegeben werden.

Die Schlüsselcodes, die dem IQTIG zum Zeitpunkt der Publikation der QS-Spezifikation bekannt sind, sind in der Tabelle `SchlüsselWert` enthalten. Spätere Schlüsseländerungen bzw. Fortschreibungen müssen (z.B. vom Softwareanbieter und von der datenentgegennehmenden Stelle) selbstständig und zeitnah über die jeweilige Bezugsquelle bezogen werden.

Die Schlüsselcodes sind in der Tabelle `SchlüsselWert` enthalten. Spätere Schlüsseländerungen bzw. -fortschreibungen werden vom IQTIG zeitnah übernommen.

Schlüsselwerte

Tabelle 21 gibt einen Überblick über die Datenbanktabelle `SchlüsselWert`, in der die Codes und Bezeichnungen der Schlüssel hinterlegt sind. Identifizierendes Merkmal ist hier eine Kombination der Spalten `fkSchlüssel` und `code`. Das bedeutet, dass jeder Schlüsselcode innerhalb eines Schlüssels nur einmal vorkommen darf.

Tabelle 21: Struktur der Tabelle `SchlüsselWert`

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

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 2.3.7) Felder mit numerischen Schlüsseln (Basistyp `NUMSCHLUESSEL`) vorkommen, so werden bei der Evaluierung der Regel die Schlüsselcodes wie ganze Zahlen behandelt.

Sortierung der Codes

- Für die Codes (Attribut `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 `Schluesssel` alphanumerisch zu sortieren.

- Spezielle Sortierung: Wenn `sortierNrVerwendet = WAHR`, so sind die Codes nach den Werten in der Spalte `sortierNr` der Tabelle `Schluesssel` numerisch zu sortieren.

Suchfunktion bei Schlüsseln mit einer großen Anzahl von Codes

Bei Schlüsseln mit einer großen Anzahl von Codes soll eine anwenderfreundliche Möglichkeit zur Auswahl der passenden Codes bereitgestellt werden. Die Umsetzung als Auswahlliste (z.B. Combobox) führt zu erhöhtem Dokumentationsaufwand, falls der Anwender über Pfeiltasten oder Schieberegler zum passenden Code navigieren muss. Ergänzend soll daher eine Suchfunktion realisiert werden, die eine Suche über die Attribute `SchluessselWert.code` oder `SchluessselWert.bezeichnung` ermöglicht. Die zu realisierenden Anwendungsfälle werden in den folgenden Beispielen erläutert.

2.2.4 Überschriften

Die Überschriften der Dokumentationsbögen in der Spezifikation sind in der Tabelle `Ab-schnitt` zu finden.

Tabelle 22: Struktur der Tabelle `Abchnitt`

| Feldname | Datentyp | Bemerkung |
|-------------------------------|----------|-------------------------------------------------------------------|
| <code>idAbschnitt</code> | INTEGER | Primärschlüssel |
| <code>bezeichnung</code> | TEXT | Text der Überschrift |
| <code>ebene</code> | INTEGER | Zeigt die Hierarchie der Überschriften an |
| <code>fkStartBogenFeld</code> | INTEGER | Fremdschlüssel auf das erste zur Überschrift gehörende Bogenfeld |
| <code>fkEndeBogenFeld</code> | 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 (z.B. 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 (Einleitung, Abschnitt 1.1). Jeder Ausfüllhinweis ist ein HTML-Dokument.

In der Spalte `ahinweis` der Tabelle `BogenFeld` ist festgelegt, welcher HTML-Ausfüllhinweis mit einem Datenfeld verknüpft ist:

`<aHinweis>.htm` = Name der HTML-Datei

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 23: Arten von Hinweistypen

| <code>fkAhinweisTyp</code> | Beschreibung | Beispiel |
|----------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| Feldbezogen | Der Ausfüllhinweis bezieht sich auf den entsprechenden Datensatz in der Tabelle <code>Feld</code> . Der Ausfüllhinweis ist modulunabhängig. | <code>IKNRKH.htm</code> Der Ausfüllhinweis bezieht sich auf verschiedene Module, beispielsweise auf <code>NWIEA</code> und <code>NWIES</code> . |
| Modulspezifisch | Soll sich ein Ausfüllhinweis nur auf ein bestimmtes Modul beziehen, kann der Attributwert <code>modulspezifisch</code> ausgewählt werden. | <code>LLPROPH\$NWIEA.htm</code> Der Ausfüllhinweis bezieht sich nur auf das Modul <code>NWIEA</code> . |
| Speziell | Soll es für verschiedene Datenfelder der Tabelle <code>Feld</code> einen gemeinsamen Ausfüllhinweis geben, kann dieser als speziell deklariert werden. Der Attributwert <code>ahinweis</code> definiert den Namen des Ausfüllhinweises. | <code>DESINFEKTINTSTAT.htm</code> Die Felder <code>DESINFEKTIONOP</code> und <code>DESINFEKTION</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 24: Tabelle `RegelTyp`

| <code>idRegelTyp</code> | bezeichnung |
|-------------------------|-------------|
| H | hart |
| W | weich |

Weiterhin wird zwischen sogenannten **Einzelregeln** (Abschnitt B 2.3.4) und **Feldgruppenregeln** (Abschnitt B 2.3.7) unterschieden.

2.3.1 Die Regeltabelle

Die Bedingungen für unplausible Angaben²⁹ 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 25: Struktur der Tabelle `Regeln`

| Feldname | Datentyp | Bemerkung |
|--------------------------------|----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>idRegeln</code> | INTEGER | Primärschlüssel |
| <code>fkModul</code> | INTEGER | Fremdschlüssel zur Tabelle <code>Modul</code> |
| <code>bedingung</code> | MEMO | Entsprechend der Syntax definierte Regeln |
| <code>meldung</code> | MEMO | Fehlermeldung: Diese Texte sind bei Regeln mit Bezug zu Feldgruppen automatisch generiert. |
| <code>alternativmeldung</code> | MEMO | Alternative Fehlermeldung: Wenn hier ein Text vorhanden ist, so ist dieser anstelle des Textes in der Spalte <code>meldung</code> zu verwenden. |
| <code>fkMehrfachregel</code> | INTEGER | Fremdschlüssel zur Tabelle mit mehrfach vorkommenden Regeln, die mithilfe von Ersatzbedingungen nach dem Export gültig sind. |
| <code>fkFeldGruppe</code> | INTEGER | Optionaler Fremdschlüssel zur Tabelle <code>Feld-Gruppe</code> : Indikator dafür, ob eine Regel aus einer Feldgruppe generiert wurde. |
| <code>fkRegelTyp</code> | TEXT(1) | Fremdschlüssel zur Tabelle <code>RegelTyp</code> : Die Regeltypen sind die in Abschnitt A 2.3 beschriebenen Arten der Plausibilitätsprüfungen: H, W oder D |
| <code>gueltigNachExport</code> | 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). |

²⁹ Eine Plausibilitätsregel müsste eigentlich „Unplausibilitätsregel“ heißen, weil sie unplausible Zustände beschreibt, die zu Fehlermeldungen führen.

Regelfelder (Bogenfelder einer Regel)

Die Tabelle `RegelFelder` (Tabelle 26) 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 26: Struktur der Tabelle `RegelFelder`

| Feldname | Datentyp | Bemerkung |
|--------------------------|----------|----------------------------------------------------------------------------------------------------------------|
| <code>fkBogenFeld</code> | INTEGER | Fremdschlüssel zu den Tabellen <code>Feld</code> und <code>Regeln</code> , bilden zusammen den Primärschlüssel |
| <code>fkRegeln</code> | 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.

Für die Module `NWIES` und `NWIEA` sind keine Mehrfachregeln definiert. Weitere Erläuterungen zu Mehrfachregeln sind der Technischen Dokumentation zur Basisspezifikation für Leistungserbringer zu entnehmen.

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 zugrunde liegende Regelsyntax wird in diesem Abschnitt beschrieben. Jede Regel ist ein logischer Ausdruck, dessen Ergebnis `WAHR` oder `FALSCH` lautet. Jede Regel bezieht sich auf einen eingegebenen Datensatz eines Moduls, dessen Daten in Variablen gespeichert sind.

Die Regelsyntax lehnt sich an die logischen Ausdrücke in bekannten Programmiersprachen an. Jedoch haben die Operatoren deutsche Namen, z.B. `UND` statt `AND` oder `ODER` statt `OR`. Die Regelsyntax ist als Pseudocode zu verstehen.

Typen

Die möglichen Typen der Datenfelder sind in Tabelle 27 aufgelistet.

Tabelle 27: Basistypen der Datenfelder in den Plausibilitätsregeln

| Basistyp | Bezeichnung | Beispiele (Literele) |
|-------------------------|--------------------------------------------------|----------------------|
| BOOL | Boolesche Variable | WAHR, FALSCH |
| TEXT | Zeichenkette (String) | "Spezifikation" |
| GANZEZAHL ³⁰ | ... -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 diese 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).

³⁰ Beim Typ GANZEZAHL sind auch negative ganze Zahlen erlaubt.



Achtung

In einer Regel dürfen nur die Feldnamen der im betreffenden Modul definierten Bogenfelder³¹ enthalten sein. Bei der Evaluierung von Regeln werden die aktuellen Werte der referenzierten Bogenfelder eingesetzt. Kann-Bogenfelder können auch un- ausgefüllt sein, also den Wert LEER haben.

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.

Literale

Alphanumerische Literale (z.B. `SCHLUESSEL`) werden von einfachen Hochkommata eingeschlossen, während Zeichenketten vom Datentyp `TEXT` in Anführungszeichen gesetzt werden müssen.³²

Dies gilt nicht für numerische Literale (`GANZEZAHL`, `ZAHL`, `NUMSCHLUESSEL` und `JAHR-DATUM`) und Literale des Datentyps `BOOL` (Wahrheitswerte).

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.

Außerdem gibt es Teildatensatz-Listenfelder, die im Abschnitt B 2.3.6 beschrieben werden.

Operatoren

Tabelle 28 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 28 hat jeder einzelne Operator eine Präzedenzstufe (höchste Präzedenzstufe ist 0). Operatoren, die die gleiche Stufe haben, werden nach den Regeln der Assoziativität aufgelöst.

Tabelle 28: 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 | |

³¹ Bei den Ersatzregeln in Tabelle `MehrFachRegel` sind stattdessen die Exportfelder des Moduls erlaubt.

³² Beim Export entfallen die begrenzenden Zeichen.

| Präzedenz | Assoziativität | Operator | Erläuterung |
|-----------|----------------|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | links | JEDESIN | 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 | 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“ |
| | 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.

Eine Besonderheit bei Listenoperationen ist die Prüfung, ob alle Elemente einer Liste ausgefüllt sind:

Beispiel:

`NICHT FELD JEDESIN (LEER)`

Diese Bedingung erfordert, dass zumindest ein Listenelement ausgefüllt ist. Beispielsweise erfüllt `FELD := (WERT;LEER;LEER;LEER)` die Bedingung.

Gleichwertig ist die Regel:

`FELD EINSNICHTIN (LEER)`

Folgende Operatoren sind komplementär:

IN und NICHTIN

EINSIN und KEINSIN

JEDESIN und EINSNICHTIN

Folgende Ausdrücke sind gleich:

A EINSNICHTIN B

NICHT A JEDESIN B

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 2.4.2).

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. 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:³³

- Befehlszeilen werden mit Semikolon abgeschlossen ;
- Wertzuweisungen mit dem Operator :=

```
A := B + C;
```

- Auswahlenweisungen

³³ Der Pseudocode erhebt nicht den Anspruch auf formale Korrektheit.

```
if (<Bedingung>){  
    ...  
}  
else {  
    ...  
}
```

Hinter <Bedingung> verbirgt sich ein logischer Ausdruck, der der Syntax der Plausibilitätsregeln gehorcht.

- Blöcke werden durch geschweifte Klammern definiert.

```
{  
...  
}
```

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

2.3.4 Syntaxvariablen

Der Eingangsdatensatz bildet die medizinische Routinedokumentation ab, die in jedem Krankenhausinformationssystem (KIS) enthalten ist. Syntaxvariablen in der QS-Dokumentation dienen der technischen Darstellung der automatischen Generierung von Angaben aus dem Eingangsdatensatz. Diese sind in Tabelle `SyntaxVariable` hinterlegt. In den Modulen `NWIES` und `NWIEA` werden derzeit keine Syntaxvariablen verwendet. Weiterführende Erläuterungen zu Syntaxvariablen sind der Technischen Dokumentation zur Basisspezifikation für Leistungserbringer zu entnehmen.

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 2.3.7 beschriebenen Feldgruppen zu unterscheiden.

Feldbezogene Prüfungen – beispielsweise Wertebereichsüberprüfungen – sind in der formalen Regelsyntax in Tabelle **Regeln** formuliert.

Unter feldbezogenen Prüfungen sind aber auch die in Abschnitt B 2.3.8 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 Angaben³⁴.

Feldübergreifende Regeln können auch teildatensatzübergreifende Regeln sein, wenn die Datenfelder der Regel aus mehreren Teildatensätzen eines Moduls stammen (Abschnitt B 2.3.6).

2.3.6 Teildatensatzübergreifende Regeln

Eine Regel ist teildatensatzübergreifend, wenn die Datenfelder der Regel aus mehreren Teildatensätzen eines Moduls stammen.

Es gibt zwei Arten von teildatensatzübergreifenden Regeln:

1. Die Felder sind in verschiedenen Teildatensätzen eines Moduls definiert.
2. Ein Feld der Regel ist in einem wiederholbaren Teildatensatz definiert und die Regel bezieht sich auf alle Werte des Datenfeldes innerhalb eines Datensatzes (= Summe aller Teildatensätze eines Vorgangs).

Regeln mit Teildatensatz-Listenfeldern

Zu jedem skalaren Datenfeld eines wiederholbaren Teildatensatzes existiert ein Teildatensatz-Listenfeld (kurz TDS-Listenfeld), das über das @-Zeichen vor dem Feldnamen angesprochen wird. Das TDS-Listenfeld enthält sämtliche Werte des betreffenden Datenfeldes, die innerhalb der QS-Dokumentation eines Falles existieren.

2.3.7 Feldgruppenregeln

Logische Abhängigkeiten von Bogenfeldern werden über Feldgruppen dargestellt. Die Plausibilitätsregeln, die einen Bezug zu einer Feldgruppe aufweisen (Tabelle Regeln), werden anhand der Feldgruppendefinition (Tabelle FeldgruppeFelder) automatisch generiert. Die Menge der abgeleiteten Einzelregeln wird in diesem Abschnitt erläutert.

Die möglichen Antworten³⁵ eines jeden Datenfeldes werden in zwei Gruppen aufgeteilt. Die erste Gruppe ist die Menge der positiven, die zweite Gruppe die Menge der negativen Antworten.³⁶

Typische positive Antworten sind beispielsweise:

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

Die komplementären negativen Antworten würden entsprechend wie folgt lauten:

³⁴ Eine Plausibilitätsregel müsste eigentlich „Unplausibilitätsregel“ heißen, weil sie unplausible Zustände beschreibt, die zu Fehlermeldungen führen.

³⁵ Die Antworten eines Datenfeldes umfassen hier neben möglichen Werten (z.B. Schlüsselwerten) oder Wertemengen auch die Kategorie „nicht ausgefüllt“ (LEER).

³⁶ Die negativen Antworten sind abhängig von der definierten Bedingung eines Feldes in der entsprechenden Feldgruppe.

Feld = LEER oder Feld NICHTIN (2 ; 3)

Eine Feldgruppe kann ein Filterfeld haben. Wenn die Antwort dieses Filterfeldes negativ ausfällt (Bspw. Bedingung: Feld = 3; Antwort: Feld <> 3), so darf keines der abhängigen Felder positiv beantwortet werden.

Tabelle 29 gibt einen Überblick über die Typen von Feldgruppen. Der aktuelle Stand findet sich in der Tabelle FeldGruppenTyp der Spezifikation.

Tabelle 29: 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.

Struktur der Tabellen `FeldGruppe` und `FeldgruppeFelder`

Die Feldgruppen sind in den Tabellen `FeldGruppe` und `FeldgruppeFelder` definiert. In der Tabelle `FeldGruppe` (Tabelle 30) sind Name, Typ und die Zuordnung zu einem Modul definiert. Die Verknüpfungstabelle `FeldgruppeFelder` (Tabelle 31) definiert die abhängigen Bogenfelder. Zusätzlich wird hier festgelegt, welche Bogenfelder der Feldgruppe als Filterfelder dienen.

Tabelle 30: Struktur der Tabelle `FeldGruppe`

| Feldname | Datentyp | Bemerkung |
|-------------------------------|-------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>idFeldGruppe</code> | INTEGER | Primärschlüssel |
| <code>name</code> | 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> | 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. |
| <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 Die generierten Einzelregeln der Feldgruppe haben den gleichen Regeltyp. |
| <code>nurPositiv</code> | BOOLEAN | Nur bei Filter-Feldgruppen wirksam: Wenn WAHR, dann umfasst die Feldgruppe nur diejenigen Regeln, welche sich auf die positive Filterbedingung beziehen. |
| <code>grauWennNegativ</code> | BOOLEAN | Definiert eine Layout-Feldgruppe, wenn WAHR (siehe unten S. 76, Layout-Feldgruppen) |

Tabelle 31: Struktur der Tabelle *FeldgruppeFelder*

| Feldname | Datentyp | Bemerkung |
|----------------------------|----------|-----------------------------------------------------------------------------------------------------------------------|
| idFeldgruppeFelder | INTEGER | Primärschlüssel |
| fkFeldGruppe | INTEGER | Obligatorischer Fremdschlüssel zur Feldgruppe |
| fkBogenFeld | INTEGER | Obligatorischer Fremdschlüssel zum Bogenfeld |
| bedingung | TEXT | Positive Bedingung für das jeweilige Bogenfeld |
| istFilter | BOOL | Legt fest, ob das jeweilige Bogenfeld ein Filterfeld ist |
| bezeichnungSchluesselListe | TEXT | Abkürzende Bezeichnung für eine Schlüsseliste in der Bedingung, wird beim Generieren von Fehlermeldungen verwendet. |
| tdsFilter | BOOLEAN | Das Bogenfeld wird in Regeln als TDS-Listefeld (Abschnitt 2.3.6) verwendet (Vorstellen des @-Zeichens vor Feldnamen). |

Syntax der Feldgruppenregeln

In den Tabellen *FeldGruppe* bzw. *FeldgruppeFelder* sind die positiven Bedingungen für das Filterbogenfeld bzw. die abhängigen Bogenfelder einer Feldgruppe definiert. Jede Bedingung hat folgenden Aufbau:

<Operator> <Operand>

Der linke Operand wird hier weggelassen, weil er immer der Name des jeweiligen Bogenfeldes ist. Die komplette Bedingung für das Bogenfeld einer Feldgruppe lautet also:

<Bogenfeld> <Operator> <Operand>

Als Operator kann jeder dyadische Operator der Tabelle 28 verwendet werden. Die auf der rechten Seite erlaubten Operanden sind nachfolgend aufgelistet:

- Literale (Tabelle 27)
- LEER
Kodelisten, in denen auch die Codes eines Schlüssels referenziert werden können; Beispiel: (1 ; 2 ; 3) oder (MaDCIS)
- ICD-Listen oder OPS-Listen (z.B. OPS_AxLKEntfOhneMark)



Hinweis

Der rechte Operand darf kein Bogenfeld sein, da sich eine Feldbedingung immer genau auf ein Bogenfeld bezieht.

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 32: 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:

A_1, A_2, \dots, A_n

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:

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

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

Feldgruppen ohne Filter

- Regeln der Feldgruppe „Einfachauswahl“ (EF)

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

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

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

- Regeln der Feldgruppe „Obligatorische Mehrfachauswahl“ (MF_MINDESTENS1)

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

Insgesamt ist eine Einzelregel mit der Feldgruppe verknüpft.

- Regeln der Feldgruppe „Und-Regel“ (UND)

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

Insgesamt ist eine Einzelregel mit der Feldgruppe verknüpft.

Feldgruppen mit mehreren Filterfeldern

Es besteht die Möglichkeit, Feldgruppen mit mehr als einem Filterfeld zu definieren:

Formal gibt es dann die Filterfelder F_1, F_2, \dots, F_n mit den positiven bzw. negativen Bedingungen $p(F_j)$ bzw. $n(F_j)$. Für alle Filterfelder wird eine positive Bedingung $p(F_1, \dots, F_n)$ und eine negative Bedingung $n(F_1, \dots, F_n)$ gebildet. Diese modifizierten

Filterbedingungen ersetzen die im Abschnitt B 2.3.7 definierten Filterbedingungen $p(F)$ und $n(F)$ bei den Einzelregeln.

Die Filterfelder können entweder über eine ODER-Verknüpfung oder eine UND-Verknüpfung miteinander verbunden sein:

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

(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.

Layout-Feldgruppen haben folgende Eigenschaften:

- Sie haben mindestens ein Filterfeld.
- Jedes abhängige Feld hat die Bedingung `<> LEER` oder `EINSNICHTIN (LEER)` (Attribut `bedingung` in Tabelle `FeldGruppeFelder`).
- Das Attribut `nurPositiv` hat den Wert `FALSCH`.

Bei Vorliegen dieser drei Eigenschaften müssen die abhängigen Felder leer bleiben, wenn die negative Filterbedingung bei der Dokumentation eines Falles erfüllt ist.

Beispiel:

Wenn in Datenfeld 73 = 0 (nein) angegeben ist, so müssen die Datenfelder 74, 75 und 76 leer bleiben. Die folgenden Plausibilitätsprüfungen stellen dies sicher.

2.3.8 Prüfung von Feldeigenschaften

Die in diesem Abschnitt behandelten feldbezogenen Prüfungen ergeben sich direkt aus den Tabellen `Feld` und `BogenFeld` 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 2.4.2): Die QS-Dokumentations- bzw. Exportsoftware muss Daten im korrekten Format generieren, was durch die datenentgegennehmenden Stellen geprüft wird.

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

Der Wert '<WERT>' des Datenfeldes <Modul.name> : <Bogen.name> : <Feld.name> '<BogenFeld.bezeichnung>' (Zeile <BogenFeld.zeileAufBogen>) ist kein gültiger <BasisTyp.name> Wert (<BasisTyp.bezeichnung> <BasisTyp.format>).

Überprüfung der Feldlänge

Die Feldlängenprüfung bezieht sich darauf, ob die Anzahl der Zeichen eines Wertes die spezifizierte Länge³⁷ (Attribut laenge in Tabelle FELD) des Feldes überschreitet.

Standardisierter Fehlertext für Längenfehler

Der Wert '<WERT>' des Datenfeldes <Modul.name> : <Bogen.name> : <Feld.name> '<BogenFeld.bezeichnung>' (Zeile <BogenFeld.zeileAufBogen>) überschreitet die zulässige Feldlänge <Feld.laenge>.

Überprüfung der Schlüsselcodes

Die Überprüfung von Schlüsselcodes bezieht sich darauf, ob bei Schlüsselfeldern nur zulässige Schlüsselcodes verwendet werden.

Standardisierter Fehlertext bei unzulässigen Schlüsselcodes

Der Wert '<WERT>' des Datenfeldes <Modul.name> : <Bogen.name> : <Feld.name> '<BogenFeld.bezeichnung>' (Zeile <BogenFeld.zeileAufBogen>) ist kein zulässiger Code des Schlüssels <Schluessel.name> (<Schluessel.bezeichnung>).

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)

³⁷ Wenn bei einem Export- oder Zusatzfeld die Länge nicht spezifiziert ist, so entfällt die Prüfung.

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.

Standardisierter Fehlertext bei Unterschreitung einer Wertebereichsgrenze

Der Wert '`<WERT>`' des Datenfeldes `<Modul.name> : <Bogen.name> : <Feld.name> ' <BogenFeld.bezeichnung> ' (Zeile <BogenFeld.zeileAufBogen>)` ist kleiner als '`<Feld.min>`'.

Standardisierter Fehlertext bei Überschreitung einer Wertebereichsgrenze

Der Wert '`<WERT>`' des Datenfeldes `<Modul.name> : <Bogen.name> : <Feld.name> ' <BogenFeld.bezeichnung> ' (Zeile <BogenFeld.zeileAufBogen>)` ist größer als '`<Feld.max>`'.

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

Das Datenfeld '`<Modul.name> : <Bogen.name> : <Feld.name> ' <BogenFeld.bezeichnung> ' (Zeile <BogenFeld.zeileAufBogen>)` muss einen gültigen Wert enthalten.

Für Exportfelder (Tabelle `ExportFormat`), die einen Bezug zu einem Ersatzfeld (Tabelle `ErsatzFeld`) bzw. zu einem Zusatzfeld (Tabelle `ZusatzFeld`) haben, sind die standardisierten Fehlertexte anzupassen:

Standardisierter Fehlertext für Muss-Fehler eines Ersatzfeldes

Das Ersatzfeld des Datenfeldes `<Modul.name> : <Bogen.name> : <Feld.name> ' <BogenFeld.bezeichnung> ' (Zeile <BogenFeld.zeileAufBogen>)` 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.`

2.3.9 Verfahren für die Evaluation von Regeln

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

1. Für mindestens ein referenziertes Bogenfeld³⁸ schlägt eine harte Feldprüfung (Abschnitt B 2.3) fehl.³⁹
2. Ein Feld der Regel ist nicht ausgefüllt (LEER) und **keine** der folgenden Teilbedingung trifft in Bezug auf das leere Feld zu:
 - Es ist in einer Liste enthalten, die mit einem Listenoperator (EINSIN, KEINSIN, JEDESIN, EINSNICHTIN) geprüft wird bzw. wird direkt gegen eine Liste geprüft (IN, NICHTIN).
 - Es wird in der Regel explizit mit <> oder = auf LEER geprüft.
 - Das Feld der Regel befindet sich auf einem vorhandenen Teildatensatz.⁴⁰
 - Jeder Operation auf einen Wert <> LEER ist eine ODER-Verknüpfte Prüfung auf LEER direkt vorgeschaltet (Feld = LEER ODER Feld Operator Operand).
3. Eine Funktion der Regel hat das Ergebnis LEER und wird in der Regel nicht explizit mit <> oder = auf LEER geprüft.

Der Algorithmus zur Evaluation einer Plausibilitätsregel ist in 3 beschrieben.

Umgang mit Laufzeitfehlern

Bei der Evaluation von Regeln können, z.B. bei einem Vergleich von LEER < WERT, Laufzeitfehler entstehen. Solche Laufzeitfehler sind bei der Evaluation zu verhindern.

Laufzeitfehler bei der Evaluation von Regeln nach den Vorgaben sind immer ein Hinweis auf Syntaxfehler in der Regel. Das Ausbleiben von Laufzeitfehlern ist noch kein Beweis für die syntaktische Fehlerfreiheit einer Regel, da es auch fehlertolerante Parser geben könnte, die beim Verlassen des definierten Wertebereichs der Regel ein FALSCH zurückgeben können. Tatsächlich liegt aber hier ein undefinierter Zustand vor, der weder FALSCH noch WAHR ist.

Ein undefinierter Zustand an einem beliebigen Punkt im Evaluationsprozess muss für alle dort noch möglichen Werte durch die Struktur der Regel immer vermieden werden.

³⁸ Die Verbindung zwischen Regeln und Bogenfeldern geschieht über die Tabelle RegelFelder, siehe Regelfelder (Bogenfelder einer Regel), Abschnitt B 1.4.1.

³⁹ Erst bei Fehlerfreiheit der feldbezogenen Prüfungen werden die feldübergreifenden Prüfungen durchgeführt.

⁴⁰ Eine Regel wird nicht geprüft, wenn der entsprechende Teildatensatz eines leeren Feldes optional und im konkreten Fall nicht vorhanden ist.

Algorithmus zur Evaluation einer Plausibilitätsregel

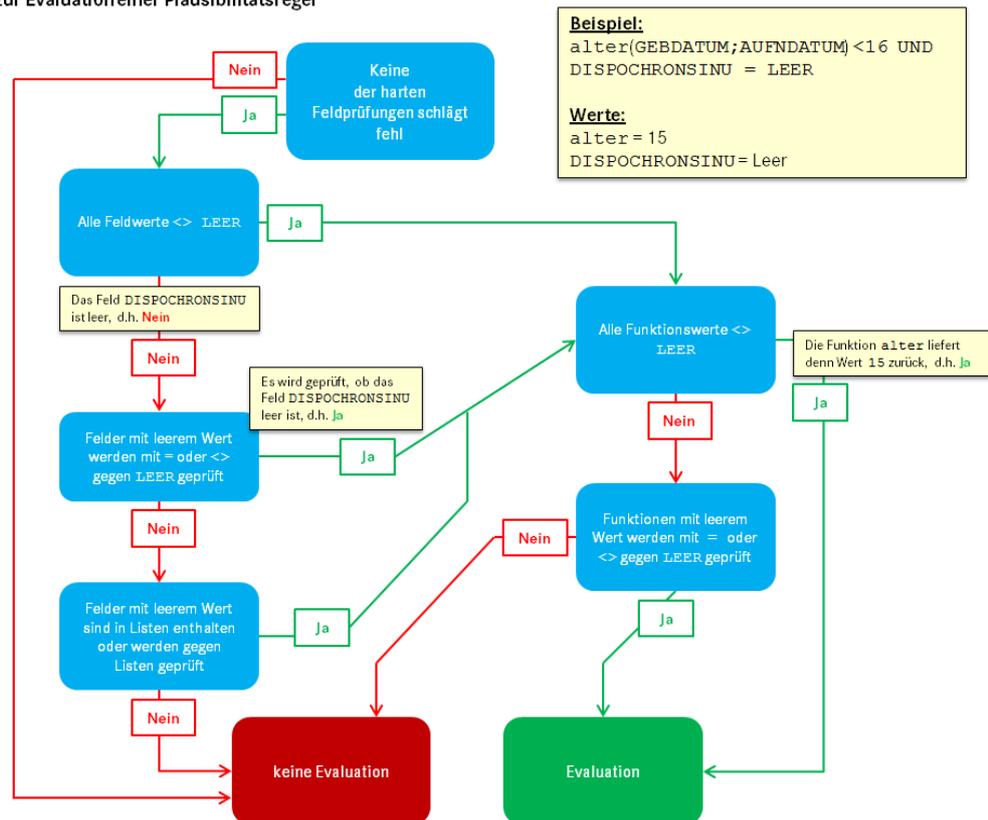


Abbildung 3: Algorithmus zur Evaluation von Plausibilitätsregeln

Teildatensatzübergreifende Regeln

Teildatensatzübergreifende Regeln (Abschnitt B 2.3.6) müssen u.U. mehrfach evaluiert werden (für jede Kombination von Teildatensätzen, die von der Regel betroffen ist).



Hinweis

In wenigen Einzelfällen beziehen sich Plausibilitätsregeln auf mehr als zwei Teildatensätze.

2.4 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. Es wird zudem zwischen Dokumentationsmodulen (Abschnitt B 2.2.1) und Exportmodulen unterschieden.

2.4.1 Exportmodule

Um die unterschiedlichen Abrechnungswege und davon abhängige Datenflüsse mit unterschiedlichen Datenannahmestellen für die Datenübermittlung zu berücksichtigen, wird zwischen den Dokumentationsmodulen (Tabelle Modul) und den Exportmodulen (Tabelle Exportmodul) unterschieden.

Tabelle 33: Struktur der Tabelle *ExportModul*

| Feldname | Datentyp | Bemerkung |
|-------------------|----------|-------------------------------------------------------------------------------------------------|
| idExportModul | INTEGER | Primärschlüssel |
| fkModul | INTEGER | Bezug zum Dokumentationsmodul |
| name | TEXT | Technischer Name (muss eindeutig sein) |
| bezeichnung | TEXT | Bezeichnung des Exportmoduls |
| existenzBedingung | TEXT | Definiert, unter welcher Bedingung das Modul in ein definiertes Exportmodul transformiert wird. |
| type_QS_data | TEXT | Datentyp im XML-Schema |

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

Beispiel:

Beim Exportmodul `PCI_LKG` ist der Datentyp im XML-Schema `qs_data_pci_type`.

Softwareanbietern soll hiermit die Integration eines Mechanismus der automatischen Datentypzuweisung ermöglicht werden, um den Aufwand zu reduzieren und Fehler zu vermeiden.

2.4.2 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. Weiterführende Informationen zum Exportdatensatz sind der Technischen Dokumentation zur Basisspezifikation für Leistungserbringer zu entnehmen.

2.5 Versionierung

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

2.5.1 Grundlegende Definitionen

In der Tabelle `Version` finden sich Informationen zur Version der Spezifikationsdatenbank. Die wichtigsten Eigenschaften einer Version sind der Versionsname (Attribut `name`) und der Gültigkeitszeitraum (Attribute `ab` und `bis`).



Achtung

Der Gültigkeitszeitraum einer Version ist in der fallbezogenen QS-Basisspezifikation für Leistungserbringer in der Regel ein Erfassungsjahr (z.B. Aufnahme zwischen dem

01.01.2017 und dem 31.12.2017). Die Auslösung der einrichtungsbezogenen QS-Dokumentation bezieht sich bei den Modulen `NWIEA` und `NWIES` nur auf **die ersten beiden Quartale des Erfassungsjahres**.

Die Spezifikation der einrichtungsbezogenen QS-Dokumentation ist beispielsweise für das Erfassungsjahr 2017 vom 01.01.2017 bis zum 31.06.2017 gültig. Die Prozesse des Verfahrens können außerhalb dieses Zeitraums liegen.

Die QS-Dokumentationssoftware eines Erfassungsjahres wird für diejenigen Einrichtungen verwendet, die Fälle mit Aufnahmedatum in das Krankenhaus (stationäre Fälle) bzw. mit Behandlungsdatum (ambulante Fälle) im oben definierten Gültigkeitszeitraum behandeln.

Jedes Modul der Datenbank hat eine Version (vgl. Attribut `fkVersion` in Tabelle `Modul`). In einer Spezifikationsdatenbank können Module unterschiedlicher Versionen enthalten sein. Über die in der Datenbank definierten Relationen sind auch für alle Bogenfelder (Tabelle `Bogenfeld`), Exportfelder (Tabelle `ExportFormat`) und Plausibilitätsregeln (Tabelle `Regeln`) Versionen definiert.

Status der Spezifikation

Versionen können den Status `in Entwicklung`, `final` oder `Update` der finalen Spezifikation haben. Diese Zustände werden zum Nachschlagen in der Tabelle `VersStatus`⁴¹ verwaltet. Das Attribut `gueltig` zeigt die gültige Version der Datenbank an. Es darf nur eine einzige Version als gültig markiert sein.

Beispiel:

Die Version 2017 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.

2.5.2 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. Weiterführende Erläuterungen zu den Delta-Informationen zur vorhergehenden Version sind der Technischen Dokumentation zur Basisspezifikation für Leistungserbringer zu entnehmen.

⁴¹ Der Status „Service Release zur finalen Spezifikation“ wird seit der Spezifikation 2014 nicht mehr verwendet.

2.6 Administrative Objekte

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

Thematisch sind diese Objekte in drei Bereiche aufgeteilt:

1. Mapping-Informationen, mit denen QS-Daten, die außerhalb des eigentlichen QS-Containers (dem Element `<qs-daten>`) im XML verortet werden können.
2. Eine Übersicht über die für ein Modul innerhalb einer Region relevanten technischen Datenservices mit Empfängeradressen und zu verwendenden XML-Schlüsseln.
3. Eine Auflistung und Kategorisierung von Prüfschritten zur Implementierung eines Datenservices.

Weiterführende Erläuterungen zu den Administrativen Objekten sind der Technischen Dokumentation zur Basisspezifikation für Leistungserbringer zu entnehmen.

3 XML-Schema

Die XML-Schema-Datei (XSD) ist eine Empfehlung des W3C⁴² zum Definieren von Strukturen für XML-Dokumente.

In der Spezifikation der einrichtungsbezogenen QS-Dokumentation 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.



Hinweis

Weiterführende Erläuterungen zu den XML-Schemata sind in der Technischen Dokumentation zur Basispezifikation für Leistungserbringer beschrieben.

Spezifische Regelungen, die lediglich im Rahmen der einrichtungsbezogenen QS-Dokumentation Anwendung finden, werden an dieser Stelle bis zur Veröffentlichung der finalen Version in Form einer entsprechenden Dokumentation beschrieben.

⁴² <http://www.w3.org/XML/Schema>

4 Tools

Die Spezifikationskomponenten, die als Hilfsprogramme Prozesse in der Qualitätssicherung unterstützen, werden in der Technischen Dokumentation zur Basisspezifikation für Leistungserbringer beschrieben. Die Hilfsprogramme der QS-Basisspezifikation für Leistungserbringer basieren auf der Programmiersprache Java. Dementsprechend ist in der entsprechenden Technischen Dokumentation ein Abschnitt enthalten, der die Installation einer Java-Laufzeit-Umgebung (JRE) beschreibt. Die Tools selbst umfassen derzeit ein Verschlüsselungsmodul und ein Datenprüfprogramm.



Hinweis

Informationen zu spezifischen Tools, die im Rahmen der einrichtungsbezogenen QS-Dokumentation Anwendung finden, werden an dieser Stelle bis zur Veröffentlichung der finalen Version in Form einer entsprechenden Dokumentation beschrieben.

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 bestimmbaren natürlichen Person zugeordnet werden können. (§3 Abs. 6a BDSG) |
| Auslösekriterien | Algorithmus zur Auslösung der Dokumentationspflicht (QS-Filter). |
| Basisspezifikation | Gemeinsame Spezifikation für stationäre 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 (§9 Qesü-RL). |
| 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 sektorenübergreifende PID-Verfahren) bis zur Datenauswertungsstelle gelangen. Die Datenflüsse sind grundsätzlich in der QSKH-RL und der Qesü-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). |

| Begriff | Beschreibung |
|------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 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). |
| Dokumentation | siehe: QS-Dokumentation |
| Einheitlicher Bewertungsmaßstab (EBM) | Vergütungssystem für die Abrechnung von Leistungen im vertragsärztlichen Bereich. |
| Einrichtung | siehe: Leistungserbringer |
| Erfassungsjahr | Das Jahr, in dem die Daten erhoben werden. Hierauf beruhen die Ergebnisse der Indikatoren. Die Kriterien für die Abgrenzung des Erfassungsjahres sind in der Spezifikation zur Sollstatistik definiert. |
| Follow-up | auch: Mehrpunktmessung, Mehrzeitpunktmessung. Versorgungsleistungen, deren Qualität (z.B. ein Behandlungserfolg) nicht einmalig, sondern zu unterschiedlichen weiteren Zeitpunkten gemessen wird. Die Messungen können in einem bestimmten oder auch in unterschiedlichen Sektoren (vgl. QSKH-RL) stattfinden. |
| Gemeinsamer Bundesausschuss (G-BA) | Oberstes Beschlussgremium der gemeinsamen Selbstverwaltung der Ärzte, Zahnärzte, Psychotherapeuten, Krankenhäuser und Krankenkassen in Deutschland. Er bestimmt in Form von Richtlinien den Leistungskatalog der Gesetzlichen Krankenversicherung (GKV) für etwa 70 Millionen Versicherte und legt damit fest, welche Leistungen der medizinischen Versorgung von der GKV erstattet werden. |
| Indikator | „Anzeiger“. Auch: Qualitätsindikator. Quantitatives Maß zur Bewertung der Erreichung eines Qualitätsziels einer bestimmten gesundheitlichen Versorgungsleistung. |
| 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). |

| Begriff | Beschreibung |
|--------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Minimaldatensatz | Wenn in der klinischen Praxis Konstellationen auftreten, in denen ein Fall durch den QS-Filter ausgelöst wird, obwohl der Datensatz des entsprechenden Leistungsbereichs nicht für die Dokumentation geeignet ist, kann ein Ersatzdatensatz mit minimalen Angaben ausgefüllt werden. |
| Missing Values | „Fehlende Werte“, z.B. fehlende Antworten und nicht auswertbare Antworten bei der Auswertung eines Fragebogens. |
| Operationen- und Prozedurenschlüssel (OPS) | Kodierungssystem für medizinische Leistungen bzw. verbindlicher Abrechnungsschlüssel zwischen medizinischen Leistungserbringern und Kostenträgern (Gesetzliche Krankenkassen). |
| Patientenidentifizierende Daten (PID) | Daten, die eindeutig einen bestimmten Versicherten identifizieren (z.B. Versichertennummer). |
| Plausibilitätsprüfung | Statistisches Verfahren, mit dem die Dokumentationsdaten auf erlaubte und/oder fehlende Werte, Widerspruchsfreiheit, Werteverteilung und bekannte Korrelationen geprüft werden. |
| Probetrieb | Erprobung eines QS-Verfahrens in einer begrenzten Anzahl von Einrichtungen. Ziel ist die Prüfung, ob die für das QS-Verfahren benötigten Daten entsprechend der vom Auftragnehmer vorgesehenen Planungen für die vorgelegten Indikatoren und Instrumente erhebbar und die Ergebnisse umsetzbar sowie für die Durchführung der Qualitätssicherung verwertbar sind. |
| Pseudocode | Programmcode, der das zugrunde liegende Prinzip eines Algorithmus beschreibt, selbst aber nicht lauffähig ist. Er dient zur Veranschaulichung, unabhängig von der konkret zu verwendenden Programmiersprache. |
| 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. |

| Begriff | Beschreibung |
|----------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 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. |
| 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 |

| Begriff | Beschreibung |
|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | 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 Erfassungsjahr zu dokumentierenden Leistungen, die vom Krankenhaus durch Konformitätserklärung schriftlich bestätigt wird. Sie gibt die zu erwartende Anzahl von Fällen in den einzelnen Leistungsbereichen (Soll) an und bildet zusammen mit der Anzahl der tatsächlich durch die Leistungserbringer gelieferten Daten (Ist) die Grundlage der Vollzähligkeitsprüfung. |
| Sozialdaten | Einzelangaben über die persönlichen und sachlichen Verhältnisse (personenbezogene Daten), die von den sozialrechtlichen Leistungsträgern zur Erfüllung ihrer gesetzlichen Aufgaben gesammelt und gespeichert werden. |
| Spezifikation | Datensatzbeschreibung. Festlegung, welche Daten für die Qualitätssicherung erhoben bzw. übermittelt werden müssen, welche Prüfalgorithmen zur Anwendung kommen (z.B. für Plausibilitätsprüfungen) und wie die QS-Auslösung operationalisiert ist. Im Rahmen der Neuentwicklung von QS-Verfahren ist die Spezifikation als das Ergebnis der informationstechnischen Aufbereitung zu betrachten. |
| Strukturierter Dialog | Instrument der Qualitätsförderung. Strukturiertes Verfahren, das Einrichtungen bei der kontinuierlichen Verbesserung von Prozessen und Qualität unterstützt. Auslöser des Strukturierten Dialogs sind rechnerische Auffälligkeiten im Ergebnis eines Qualitätsindikators im Rahmen der statistischen Basisprüfung der übermittelten QS-Daten (QSKH-RL). Abweichend von der Definition in der QSKH-RL bezeichnet der Begriff „Strukturierter Dialog“ in der Qesü-RL nur den Prozessschritt, der dem schriftlichen Teil der Durchführung von QS-Maßnahmen folgt. |
| Systempflege | Routinemäßige und kontinuierliche Evaluation und Anpassung der Qualitätsindikatoren, der Softwarespezifikation usw. |
| Verfahren | siehe: Leistungsbereich |
| Vertrauensstelle | Institution, die im Rahmen der sektorenübergreifenden Qualitätssicherung erhobene patientenidentifizierende Daten pseudonymisiert. Näheres regeln die Qesü-RL (§11), QSKH-RL (Anl. 2, §4) sowie themenspezifische Bestimmungen. |
| Verweildauer | Dauer des stationären Aufenthalts eines Patienten, Abstand zwischen Aufnahme- und Entlassungsdatum. |
| Vollständigkeit | Erfassung aller zu einem einzelnen Behandlungsfall erforderlichen Angaben (Daten). |

| Begriff | Beschreibung |
|------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Vollzähligkeit | Erfassung aller dokumentationspflichtigen Behandlungsfälle zu einer bestimmten Versorgungsleistung. |
| Vollzähligkeitsprüfung | Abgleich der laut QS-Filter-Software zu erwartenden Anzahl von Fällen in einem Leistungsbereich (Soll) mit der Anzahl der tatsächlich durch die Leistungserbringer gelieferten Daten (Ist). |
| XSLT | Extensible Stylesheet Language Transformations. Programmiersprache zur Transformation von XML-Dokumenten in andere XML-Dokumente oder andere Dokumentformate wie HTML. Im QS-Kontext kann es auch für Datenprüfung und Protokollerstellung verwendet werden. |
| Zählleistungsbereich | Bei Leistungsbereichen, die über einen gemeinsamen Dokumentationsbogen erfasst werden (beispielsweise Herzchirurgie), dient der Zählleistungsbereich der Zuordnung zu einer definierten Teilmenge eines Leistungsbereichs (beispielsweise kathetergestützte vs. offen-chirurgische vs. kombinierte Eingriffe im Leistungsbereich <i>Aortenklappenchirurgie</i>). |