

# **Technische Dokumentation zur GEKID-Spezifikation (nach dem Standard der BQS-Spezifikation)**

Stand: 22.01.2009

Version: 01, gültig ab 01.01.2009

## Ansprechpartner

Epidemiologisches Krebsregister NRW gGmbH  
Robert-Koch-Str. 40  
48149 Münster

E-mail [volkmar.mattauch@krebsregister.nrw.de](mailto:volkmar.mattauch@krebsregister.nrw.de)

Das Epidemiologische Krebsregister NRW gGmbH übernimmt keinerlei Haftung für Fehler und Irrtümer, die in dieser Spezifikation enthalten sind.

# Inhaltsverzeichnis

<b>1</b>	<b>EINLEITUNG .....</b>	<b>4</b>
1.1	PROJEKTHINTERGRUND	4
1.2	BEZUG ZUR BQS-SPEZIFIKATION	4
1.3	LIEFERUMFANG	4
1.4	ALLGEMEINE ANMERKUNGEN ZUR STRUKTUR DER SPEZIFIKATIONSDATENBANK	5
1.4.1	Abfragen der Datenbank	5
1.4.2	Tabellenstruktur der Datenbank	6
<b>2</b>	<b>DATENEXPORT IM GEKID-VERFAHREN.....</b>	<b>9</b>
2.1	ÜBERBLICK	9
2.2	REGISTRIERUNG EINES DOKUMENTATIONSSYSTEMS	10
2.3	IDENTIFIZIERUNG VON DATENSÄTZEN	11
2.4	DER EXPORTVORGANG	12
2.4.1	Erzeugung der Transaktionsdatei	14
2.4.2	Die Steuerdatei	14
2.4.3	Export von Teildatensätzen	16
2.5	REGELN FÜR DIE ENTGEGENNAHME VON DATENSÄTZEN UND TEILDATENSÄTZEN	19
2.5.1	Prüfungen und Datenrückbestätigung	19
2.5.2	Stornierung von Datensätzen	21
2.6	ANTWORTDATEIEN	21
2.6.1	Die Bestätigungsdatei	22
2.6.2	Die Fehlerdatei	23
A	REFERENZEN	27

# 1 Einleitung

## 1.1 Projekthintergrund

Die Meldung bösartiger Neubildungen an die Epidemiologischen Krebsregister ist in Deutschland inzwischen in allen Bundesländern per Landesgesetze geregelt. In den meisten Bundesländern wurde diesbezüglich eine gesetzliche Meldepflicht eingeführt. Die Gesellschaft der Epidemiologischen Krebsregister in Deutschland e.V. (GEKID) hat Mitte 2008 einen Mindestdatensatz definiert, der allen Meldungen zugrunde zu legen ist (siehe [www.gekid.de](http://www.gekid.de)).

In der vorliegenden GEKID-Spezifikation werden sämtliche Merkmale des GEKID-Mindestdatensatzes so definiert, dass sie Eingang in die KIS bzw. in die QS-Systeme finden können.

Die Weiterleitung der auf diesem Wege erzeugten Transaktionsdateien muss mit dem jeweiligen Landeskrebsregister vereinbart werden. Meldungen auf der Grundlage des Mindestdatensatzes der GEKID werden aber prinzipiell von allen Landeskrebsregistern entgegengenommen.

## 1.2 Bezug zur BQS-Spezifikation

Struktur und Aufbau der Spezifikation für die Tumordokumentation (kurz GEKID-Spezifikation) basieren auf der BQS-Spezifikation für QS-Dokumentationssoftware 12.0.

## 1.3 Lieferumfang

Die vorliegende Spezifikation besteht aus folgenden Teilen:

- Technische Dokumentation: *TechDok-gekid.pdf*
- Datenbank: *spez-gekid-2009.mdb*

- ZIP-Archiv mit Ausfüllhinweisen: *Ausfuellhinweise- gekid -2009.zip*
- Dokumentationsbogen zur Visualisierung des Datensatzes: *DokuBogen-gekid-2009.pdf*

## 1.4 Allgemeine Anmerkungen zur Struktur der Spezifikationsdatenbank

Die Spezifikation ist in einer relationalen Datenbank abgelegt. Zurzeit wird sie ausschließlich als Access-Datenbank (MS Access 2000) zur Verfügung gestellt.

Der Name der Spezifikation richtet sich nach folgendem Schema:

`spez-<Version>.mdb`

`<Version>` bezeichnet die Versionskennung (z. B. `gekid-2009`).

### Beispiel:

Im Erfassungsjahr 2009 ist die Spezifikation `spez-gekid-2009.mdb` gültig.

Service-Releases zur Spezifikation sind am Namen erkennbar:

`spez-gekid-<Version>-SR<n>.mdb`

`<n>` bezeichnet die laufende Nummer (1, 2, 3) des Service-Releases.

### Beispiel:

`spez-gekid-2009-SR1.mdb` ist das erste Service-Release zur Spezifikation 2009

Eine Übersicht über die in der jeweiligen Version enthaltenen Module (Leistungsbereiche) liefert die Abfrage *Datensätze* der Access-Datenbank (siehe Kapitel 1.4.1).

### 1.4.1 Abfragen der Datenbank

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

- *Datensätze:*  
Überblick über die in der Spezifikation enthaltenen Module.

- *GEKID\_Datenfeldbeschreibung*  
Tabellarischer Überblick über die Datenfelder, sortiert nach Teildatensätzen und Zeilennummern der Bogenfelder.
- *GEKID\_Plausibilitätsregeln*  
Überblick über die formale Bedingungen und die Fehlertexte
- *GEKID\_Teildatensätze*  
Überblick über die Teildatensätze des GEKID-Datensatzes und die Regeln für das Anlegen von Teildatensätzen
- *GEKID\_Exportfelder*  
Überblick über die Exportfelder. Man erhält eine Übersicht über die zu exportierenden Felder incl. Zuordnung zum Teildatensatz
- *GEKID\_Feldgruppen*  
Übersicht über alle Feldgruppen.
- *GEKID\_WertebereicheNumerischerFelder*  
Anzeige der numerischen Datenfelder (Typ ZAHL und GANZEZAHL) und ihrer Wertebereiche.
- *GEKID\_Überschriften*  
Anzeige der Überschriften. Angegeben werden Start- und Ende-Felder der Überschriften, sowie die Ebene der Überschriften.
- *GEKID\_Ausfüllhinweise*  
Anzeige der Zuordnung von Ausfüllhinweisen (htm-Dateien) zu den Datenfeldern.
- *Schlüsselcodes*  
Zeigt alle Schlüssel und die zugehörigen Schlüsselwerte an.

#### 1.4.2 Tabellenstruktur der Datenbank

Die Tabellen und Spalten (Attribut) 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.

Beispiel:

BasisTyp (Tabelle)

`idBasisTyp (Spalte)`

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

`id<TabellenName>`

Der Ausdruck in den eckigen 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 Attribut*<sup>1</sup>, welches durch Setzen eines weiteren, eindeutigen Index (bestehend aus einem oder mehreren Attributen) definiert ist.

#### Beispiele:

- Identifizierendes Attribut: Attribut `name` in Tabelle `BasisTyp`
- Identifizierende Attributkombination: Attribute `code` und `fkSchluessel` in Tabelle `SchlüsselWert`

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

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

`fk<FremdTabellenName>`

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

Die Fremdschlüsselattribute (Namen beginnen mit `fk`) wurden in MS Access als Datenbankattribute zum Nachschlagen eingerichtet. Z.B. wird beim Fremdschlüsselattribut

---

<sup>1</sup> oder eine identifizierende Attributkombination, die einen eindeutigen Schlüssel definiert

`fkModul` in der Tabelle `Bogen` nicht mehr der Primärschlüssel des jeweiligen Moduls, sondern der Name des Moduls angezeigt.<sup>2</sup> Sind zwei Tabellen mehrfach durch Schlüssel-Fremdschlüssel-Beziehungen miteinander verknüpft, so kann der Name eines Fremdschlüssels auch folgendermaßen aufgebaut sein:

`fk<FremdTabellenName><Rolle>`

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

N-M-Beziehungen werden wie üblich über Verknüpfungstabellen realisiert. In der Spezifikation haben Verknüpfungstabellen gewöhnlich keinen Primärschlüssel, jedoch einen eindeutigen Schlüssel, der über die Fremdschlüsselfelder definiert ist. Ein Beispiel hierfür ist die Tabelle `RegelFelder`, welche die Tabellen `BogenFeld` und `Regel` verknüpft.

Folgende Attribute treten in vielen Tabellen auf und seien hier kurz erläutert:

- `name` ist in der Regel als "technischer Name" zu verstehen. Z.B. wird `Feld.name` als Variablenname in den Plausibilitätsregeln verwendet.
- `bezeichnung` ist eine kurze Beschreibung. Z.B 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`.

---

<sup>2</sup> Diese Änderung betrifft nur die Anzeige, nicht jedoch die Struktur der Datenbank!

## 2 Datenexport im GEKID-Verfahren

### 2.1 Überblick

In diesem Abschnitt wird ein Überblick über den Datenexport des BQS-Verfahrens gegeben.

Die QS-Software (QSD) stellt die modulübergreifende Funktionalität Datenexport bereit. Der Datenexport wird üblicherweise nicht durch die für die Dokumentation verantwortlichen ärztlichen oder pflegerischen Mitarbeiter, sondern durch einen Mitarbeiter der Krankenhaus-EDV durchgeführt. Exportiert werden ausschließlich diejenigen Datensätze, welche von den ärztlich-pflegerischen Mitarbeitern im Rahmen eines Dokumentationsabschlusses<sup>3</sup> freigegeben sind.

Die von der QS-Software generierten Transaktionsdateien (siehe Abschnitt 2.4) werden durch eine von der GEKID entwickelte Software weiterverarbeitet. Diese Software überträgt die Inhalte der Transaktionsdateien in die so genannte GEKID-Schnittstelle, die von den Epidemiologischen Krebsregistern entgegengenommen wird. Die genauen Modalitäten der Übermittlung müssen mit dem jeweils zuständigen Landeskrebsregister vereinbart werden.

Tabelle 1 Datenexport - Unterschiede zwischen dem QS-Verfahren und dem EKR-Verfahren

Schritt	QS-Verfahren	EKR-Verfahren
Zu exportierenden Datensätze auswählen	QSD	QSD
Exportdateien erzeugen: <ul style="list-style-type: none"><li>• Headerdateien</li><li>• Moduldateien</li></ul>	QSD	QSD
Transaktionsdatei erzeugen	QSD	
Transaktionsdatei verschlüsseln	QSD	GEKID-Software
Transaktionsdatei an Datenstelle übermitteln	QSD	GEKID-Software
Antwortdatei der Datenstelle entschlüsseln	QSD	GEKID-Software

<sup>3</sup> Eine Dokumentation kann nicht abgeschlossen werden, wenn harte Plausibilitätsregeln verletzt werden.

Schritt	QS-Verfahren	EKR-Verfahren
seln und entpacken		
Entschlüsselte Antwortdateien der Datenstelle einlesen	QSD	QSD

Softwareanbieter, welche diese Spezifikation umsetzen, stellen zwei Funktionalitäten bereit:

- Export der Transaktionsdateien in das Dateisystem
- Einlesen der Antwortdateien aus dem Dateisystem

## 2.2 Registrierung eines Dokumentationssystems

Jedes Dokumentationssystem nach dem BQS-Standard kann Daten an mehrere datenentgegennehmende Stellen übermitteln. Beispielsweise werden Daten für die Direktverfahren an die BQS und für die indirekten Verfahren an die zuständige Landesgeschäftsstelle übermittelt. Werden Daten zusätzlich an das epidemiologische Krebsregister übermittelt, so müssen die Krankenhäuser ihre Dokumentationssysteme beim Krebsregister registrieren lassen.

Jedes registrierte Dokumentationssystem bekommt vom zuständigen Krebsregister eine Registriernummer zugewiesen, welche folgendermaßen aufgebaut ist:

<Registriernummer> = <KodeDatenstelle>-<Registrierkode><sup>4</sup>

Der Kode der Datenstelle „Krebsregister“ setzt sich zusammen aus dem Kürzel KR und dem amtlichen Länderkode (Tabelle 2). Die maximal 12-stelligen Registrierkodes werden von den Krebsregistern gepflegt.

Tabelle 2 Kodes der Krebsregister

Kode der Datenstelle	Bundesland
KR01	Schleswig-Holstein
KR02	Hamburg
KR03	Niedersachsen

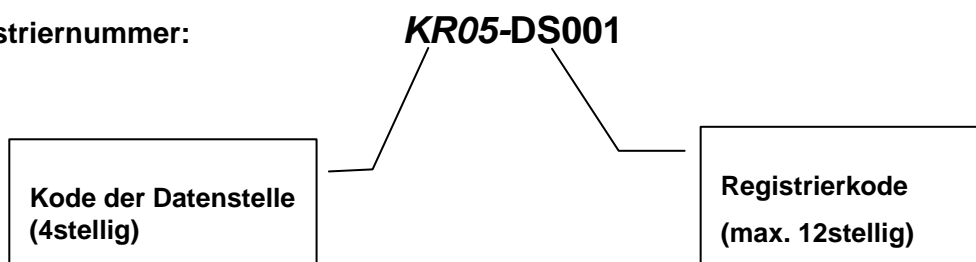
<sup>4</sup> Die Registriernummer ist maximal 17 Zeichen lang.

Kode der Datenstelle	Bundesland
KR04	Bremen
KR05	Nordrhein-Westfalen
KR06	Hessen
KR07	Rheinland-Pfalz
KR08	Baden-Württemberg
KR09	Bayern
KR10	Saarland
KR11	Berlin
KR12	Brandenburg
KR13	Mecklenburg-Vorpommern
KR14	Sachsen
KR15	Sachsen-Anhalt
KR16	Thüringen

Beispiel:

Das epidemiologische Krebsregister NRW hat den Kode *KR05*. Die in den Krankenhäusern installierten Dokumentationssysteme seien in diesem Beispiel fortlaufend durchnummeriert (DS001, DS002 etc.)<sup>5</sup>.

Registriernummer:



### 2.3 Identifizierung von Datensätzen

Die Vorgangsnummer kennzeichnet in eindeutiger Weise jeden dokumentierten Datensatz eines Dokumentationssystems, und zwar unabhängig vom angewandten Modul.

Im einfachsten Fall könnte also die Vorgangsnummer um 1 erhöht werden, wenn ein neuer Datensatz angelegt wird. Wenn während eines stationären Aufenthaltes zwei QS-

---

<sup>5</sup> Fiktives Beispiel

Dokumentationen eines Krankenhausfalls angelegt werden, so müssen auch unterschiedliche Vorgangsnummern vergeben werden.

Insbesondere ist es falsch, einfach eine Patientenidentifikationsnummer oder die Krankenhaus-Fallnummer zu verwenden bzw. zu pseudonymisieren<sup>6</sup>. Bei der Umsetzung hat der Softwareanbieter weit gehende Freiheit, vorausgesetzt, dass die modulübergreifende Eindeutigkeit der Vorgangsnummer gewährleistet ist.

Die Vorgangsnummer, welche als Zusatzfeld mit jedem Teildatensatz an die entgegennehmende Stelle übermittelt wird, darf für die Datenstelle nicht personenbeziehbar sein. In der Vorgangsnummer sollte z.B. nicht das Geburtsdatum enthalten sein.

Die Dokumentationssoftware verwaltet jahrgangsübergreifend die Vorgangsnummern der QS-Dokumentationen. Sie soll dem Krankenhaus eine Zuordnung der Vorgangsnummern zu krankenhauses internen Fall- oder Patientennummern (vgl. nicht übermitteltes Datenfeld IDNRPAT) ermöglichen.

#### *Annahme oder Ablehnung von unterschiedlichen Versionen eines Datensatzes*

Bei der entgegennehmenden Stelle eingehende Datensätze werden anhand der Kombination aus Registriernummer und Vorgangsnummer als *ein Vorgang* identifiziert. Der für den Vorgang gespeicherte Datensatz kann durch eine neuere Version (mit höherer Versionsnummer) überschrieben werden.<sup>7</sup>

Unterschiedliche Versionen eines Datensatzes müssen sich auf denselben Behandlungsabschnitt beziehen. Wenn z.B. ein Patient zwei stationäre Aufenthalte in demselben Krankenhaus hatte, so sind zwei Dokumentationen mit unterschiedlichen Vorgangsnummern anzulegen.

Ebensowenig darf eine Vorgangsnummer, welche das Dokumentationssystem bereits für eine onkologische Dokumentation verwendet wurde, für ein Modul des BQS-Verfahrens wiederverwendet werden.

## **2.4 Der Exportvorgang**

Bei einem Exportvorgang wird von einem Dokumentationssystem eine Transaktionsdatei für die jeweilige datenentgegennehmende Stelle erzeugt.

---

<sup>6</sup> Denn für jeden Patienten können unter Umständen mehrere Datensätze angelegt werden.

<sup>7</sup> Ggf. ist der geänderte Datensatz mit einer neuen Versionsnummer zu übermitteln.

Die Daten der zu exportierenden Dokumentationen werden vom Dokumentationssystem in Exportdateien geschrieben und die entsprechenden Vorgänge (identifiziert durch Vorgangsnummern) im Dokumentationssystem als "exportiert" markiert.

Jeder Exportvorgang erfolgt gemäß dem jeweils gültigen Exportverfahren. Jeder Übermittlung an eine datenentgegennehmende Stelle ordnet das registrierte Dokumentationssystem eine *eindeutige Transaktionsnummer* zu. Die Eindeutigkeit der Transaktionsnummer muss über die Erfassungsjahre hinweg gewährleistet sein.

#### Achtung:

Es ist insbesondere nicht erlaubt, für den erneuten Export korrigierter Datensätze die gleiche Transaktionsnummer wie für den ersten Export zu verwenden.

#### Nummernkreis der Transaktionsnummern

Die Transaktionsnummer ist als eindeutige, fortlaufende, positive und ganze Zahl zu wählen: 1, 2, 3 etc.

Pro Transaktion werden eine Steuerdatei (Abschnitt 2.4.2) und teildatensatzspezifische Exportdateien (Abschnitt 2.4.3) erzeugt, welche in ein ZIP-Archiv (Abschnitt 2.4.1) gepackt werden.

Üblicherweise werden die Dateien einer Transaktion von der Erfassungssoftware nach erfolgreicher Durchführung des Exportes in ein separates Verzeichnis geschrieben (Abbildung 1).

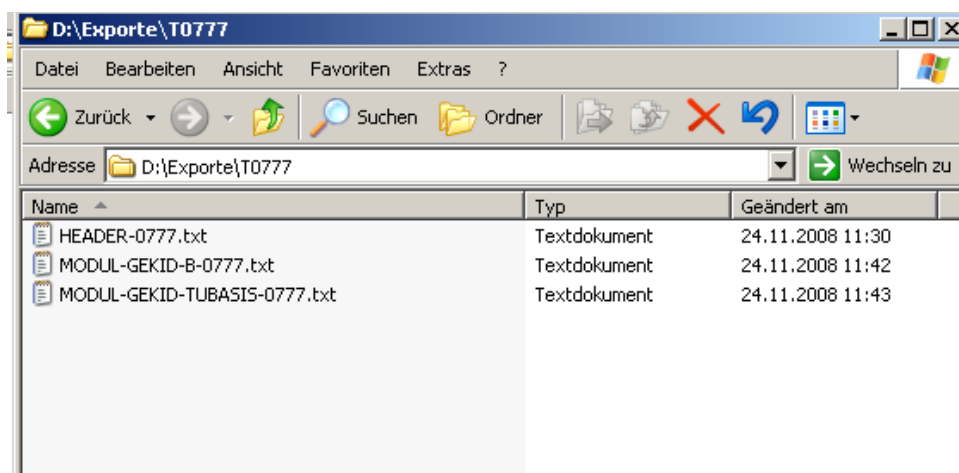


Abbildung 1 Beispiel für Aufbau eines Transaktionsverzeichnisses

## 2.4.1 Erzeugung der Transaktionsdatei

Die Steuerdatei (Kapitel 2.4.2) und die zugehörigen Exportdateien (Kapitel 2.4.3) werden vor dem Export mit Hilfe eines pkzip 2.04g-kompatiblen Archivierungsprogramms komprimiert. Die komprimierte Datei hat dann folgenden Dateinamen:

```
T-<KodeDatenstelle>-<Registrierkode>-<Transaktionsnummer>.ZIP
```

### Beispiel:

```
T-KR05-DS001-777.ZIP
```

## 2.4.2 Die Steuerdatei

Die Steuerdatei - das Inhaltsverzeichnis eines Exportvorganges - wird nach folgendem Schema benannt:

```
HEADER-<Transaktionsnummer>.txt8
```

### Beispiel:

```
HEADER-0001.txt
```

```
HEADER-0999.txt
```

Die erste Zeile der Steuerdatei besteht aus folgenden Feldern:

- Verfahrenskennung GEEKID-2009
- Version des Datenexportverfahrens (Vgl. Attribut `name` in der Tabelle `Version`)
- Kennung der Software, mit der die Exportdateien erzeugt wurden
- Datum und Uhrzeit der Erstellung der Exportdatei
- Institutionskennzeichen des Absenders (Krankenhaus)
- Registriernummer des Dokumentationssystems (Vergleiche Kapitel 2.2)
- Namenszeichen des Ansprechpartners zu Fragen der Datenübermittlung.

Jede nachfolgende Zeile spezifiziert eine Exportdatei und setzt sich aus folgenden Feldern zusammen:

- Modulbezeichnung (Attribut `name` in Tabelle `Modul`)
- Name des Teildatensatzes (Attribut `name` in Tabelle `Bogen`)

---

<sup>8</sup> Den Transaktionsnummern können führende Nullen vorangestellt werden. Z.B. 0001 statt 1

- Versionsnummer der gültigen Spezifikation (vgl. Attribut `name` in Tabelle `Version`)
- Name der Datei (zurzeit fest vorgegeben) und
- Anzahl der Datensätze in diesem Modul.

Die Felder sind durch Semikola voneinander getrennt. Jede Zeile wird durch `<CR><LF>` (ASCII 13, 10) abgeschlossen.

### Beispiele:

Steuerdatei des Verfahrens GEKID:

```
Gekid-2009;Gekid-2009;Gekid-2009#SR1#QS-EasyDoc#7.5;11.11.2009
11:11:11;261800267;DS001;OE<CR><LF>
GEKID;B;Gekid-2009;MODUL-GEKID-B-0777.txt;33<CR><LF>
GEKID;B;Gekid-2009; MODUL-GEKID-TUBASIS-0777.txt;34<CR><LF>
..
```

### Achtung:

Es ist verboten, Exportdateien unterschiedlicher Verfahrenskennungen und unterschiedlicher Versionen in einer einzigen Transaktion zu übermitteln.

### *Softwarekennung*

Die Softwarekennung setzt sich aus folgenden Teilen zusammen:

- Version der Spezifikation, auf deren Basis die QS-Dokumentationssoftware entwickelt wurde (siehe Attribut `name` in Tabelle `Version`)
- Releasekennung der Spezifikation, auf deren Basis die QS-Dokumentationssoftware entwickelt wurde (siehe Attribut `name` in Tabelle `Version`)
- Name des Softwareproduktes
- Releasekennung des Softwareproduktes

Die Teile der Softwarekennung sind durch das Zeichen `#` voneinander getrennt:

```
<Version Spez.>#<Release Spez.>#<Name Software>#<Release Software>
```

### Beispiele:

```
GEKID-2009##QSEASYDOC#001
GEKID-2009#SR1#QSEASYDOC#099
```

Exporte, deren Softwarekennung nicht dem Format genügt, sind fehlerhaft (Fehlerart STEUER):

Standardisierter Fehlertext:

Softwarekennung in Steuerdatei fehlerhaft: Angabe im Format <Version Spez.>#<Release Spez.>#<Name Software>#<Release Software> erforderlich!
--

### 2.4.3 Export von Teildatensätzen

Beim Export einer QS-Dokumentation durch ein Dokumentationssystem werden die Inhalte der für den betreffenden Vorgang angelegten Teildatensätze in passende Exportdateien geschrieben. Jede Exportdatei ist einem in der Spezifikation definierten Teildatensatz (vgl. Tabelle Bogen) zugeordnet.

Die Vorschriften zur Anonymisierung und Pseudonymisierung entfallen in der GEKID-Spezifikation.

Die Exportdateien sind wie folgt aufgebaut:

- ASCII-Format, erweiterter IBM-PC-Zeichensatz (8 Bit, DOS/OS/2),
- die erste Zeile enthält die Feldnamen, die nachfolgenden Zeilen enthalten die Werte der Exportfelder eines Teildatensatzes,
- Zeilen werden durch <CR><LF> (ASCII 13, 10) abgeschlossen,
- Feldnamen in der ersten und Werte der Exportfelder in den folgenden Zeilen werden durch Semikola getrennt und können variabel lang sein<sup>9</sup>.

#### *Felder der Exportdatei*

Einen Überblick über die zu exportierenden Felder eines Moduls liefert die Abfrage *GEKID\_Exportfelder*. Am Anfang eines jeden Teildatensatzes findet man die Zusatzfelder. Danach folgen die Bogenfelder des Teildatensatzes, wobei die zu anonymisierenden Bogenfelder durch die Ersatzfelder ersetzt sind.

#### *Format der Exportfelder*

---

<sup>9</sup> Hinter das letzte Feld darf kein Semikolon angehängt werden

In Abhängigkeit vom Basistyp des Exportfeldes werden die Werte der Exportfelder exportiert. Die Basistypen, deren Literale in Regeln von doppelten Anführungszeichen oder Hochkommata umschlossen werden, werden ohne diese Zeichen exportiert (TEXT, SCHLUESSEL, DATUM, MONDATUM, JAHRDATUM und UHRZEIT).

Folgende Grundsätze für das Format von Werten der Exportfelder sind nachfolgend noch einmal explizit aufgelistet:

- Semikola dürfen innerhalb eines Exportfeldes nicht vorkommen,
- Dezimaltrennzeichen ist das Komma, es werden keine Tausendertrennzeichen verwendet,
- Datumentrennzeichen ist der Punkt (Datumsangaben sind immer 10-stellig)
- Uhrzeittrennzeichen ist der Doppelpunkt,
- Der Code nach ICD-10-GM wird mit Trennzeichen übermittelt.

#### Beispiel:

Dieses Beispiel zeigt den Anfang einer Exportdatei für den Teildatensatz GEKID:B. Es sind nur die ersten 11 Felder dargestellt:

```
RegistrierNr;Vorgangsnr;VersionNr;Storno;Modul;Bogen;DokAbschlDat;.. <CR><LF>  
KR05-001;4711;1;;GEKID;B;31.01.2008;.. <CR><LF>
```

#### *Zusatzfelder des Datenexports*

Zusatzfelder, welche nicht in der Datenfeldbeschreibung (Tabelle `BogenFeld`) eines Moduls enthalten sind, werden von der QS-Dokumentationssoftware ausgefüllt<sup>10</sup>.

Einige der in der Tabelle `ZusatzFeld` definierten Zusatzfelder werden nachfolgend erläutert:

- Das übertragene Speicherdatum `DokAbschlDat` (*Datum des Dokumentationsabschlusses bzw. der Freigabe des Datensatzes für den Export*) ist nicht Teil der Datenbank für Auswertungen und wird nur für organisatorische Zwecke verwendet.
- Die *Versionsnummer* (`VersionNr`) gibt an, die wievielte Version des Datensatzes übertragen wird.

#### Erläuterung:

---

<sup>10</sup> Hier gilt also nicht der Grundsatz, dass Felder nicht vorbelegt sein dürfen.

In der Regel wird die Versionsnummer 1 lauten. D.h., dass der nach dem ersten Dokumentationsabschluss freigegebene Datensatz übertragen wird. Muss ein korrigierter Datensatz erneut eingesandt werden, so muss die Versionsnummer vom dokumentierenden System um eins erhöht werden. Die neue Version des Datensatzes wird bei der Entgegennahme geprüft und überschreibt bei Korrektheit die alte Version des Datensatzes.

Achtung:

Wenn die entgegennehmende Stelle einen Datensatz mit derselben Versionsnummer ein zweites Mal erhält, so wird dieser zurückgewiesen.

- Der Eintrag 1 im Zusatzfeld *Storno* veranlasst die datenentgegennehmende Stelle, den übermittelten Datensatz einschließlich seiner Vorversion(en) als „storniert“ zu kennzeichnen.

Eine vollständige Liste der möglichen Zusatzfelder findet sich in der Tabelle *ZusatzFeld* der Spezifikationsdatenbank.

*Benennung der Exportdateien*

Exportdateien sollen nach folgendem Schema benannt werden

MODUL-<Modulname>-<Teildatensatzname>-<Transaktionsnummer>.txt

Erläuterung:

- <Modulname> ist der Name des Moduls (Attribut *name* der Tabelle *Modul*), wobei der Schrägstrich durch N ersetzt wird
- <Teildatensatzname> ist in Attribut *name* der Tabelle *Bogen* zu finden.
- <Transaktionsnummer> : 1, 2, .., 99, 100, .., 999, 1000, 1001, ..

Beispiele:

MODUL-GEKID-B-0777.txt

MODUL-GEKID-TUBASIS-0777.txt

## 2.5 Regeln für die Entgegennahme von Datensätzen und Teildatensätzen

### 2.5.1 Prüfungen und Datenrückbestätigung

Wenn der Datensatz eines Vorgangs aus mehreren Teildatensätzen besteht, so wird der Datensatz des Moduls akzeptiert, wenn alle erforderlichen Teildatensätze fehlerfrei sind.

Die entgegennehmende Stelle kann nach der Prüfung eines Datensatzes zu folgenden Ergebnissen (*Bestätigungsstatus*) kommen:

- OK  
Der Datensatz ist fehlerfrei.
- FEHLER  
Der Kerndatensatz enthält Fehler und ggf. der optionale Datensatz bzw. ein Storno schlägt fehl.
- STORNO  
Der Datensatz (inkl. aller Teildatensätze) wurde von der daten-entgegennehmenden Stelle storniert.

#### Achtung:

*Bei jeder Übermittlung vom Krankenhaus an die entgegennehmende Stelle muss der gesamte Datensatz<sup>11</sup> neu übermittelt werden (auch wenn er vorher schon bestätigt worden ist)!*

#### *Definitionen der Fehlerfreiheit von Teildatensätzen und Datensätzen*

- Ein Teildatensatz ist dann fehlerfrei, wenn keines seiner Bogenfelder eine Wertebereichsverletzung aufweist und keine der verletzten Plausibilitätsregeln des Moduls einen Bezug zu einem Bogenfeld des Teildatensatzes hat.
- Der Datensatz ist dann fehlerfrei, wenn seine Teildatensätze fehlerfrei sind.

*Folgende Prüfungen beziehen sich auf den Exportdatensatz und werden von den daten-entgegennehmenden Stellen durchgeführt:*

---

<sup>11</sup> Der komplette Datensatz mit allen Teildatensätzen

*Prüfung auf Lesbarkeit und Virenfreiheit* des Datenträgers und der übermittelten Transaktionsdateien. Hier erfolgt keine Protokollierung in Form von Protokoll- und Fehlerdateien, sondern der Absender erhält eine entsprechende formlose Nachricht.

Die *formale Prüfung der Steuerdatei* (Kapitel 2.4.2) umfasst die Korrektheit des Formats.

Die *formale Prüfung der Exportdateien* umfasst die Existenz der in der Steuerdatei aufgelisteten Exportdateien und die Korrektheit des Formats.

Die *Vollständigkeits- und Versionsprüfung* bezieht sich auf einen übermittelten Datensatz, der durch Registriernummer, Vorgangsnummer und Versionsnummer identifizierbar ist:

- Zulässigkeit der Versionsnummer: wurde der Datensatz bereits übermittelt, muss die Versionsnummer des neu übermittelten Datensatzes größer als die des bereits gespeicherten Datensatzes sein.
- Teildatensätze mit dem Attribut `fkBogenZahl` + oder 1 müssen (bezogen auf den Mutterteildatensatz) mindestens einmal oder genau einmal in einem Datensatz existieren (Fehlerart TDS in Kapitel 2.6.2).
- Für mehrfach angelegte Teildatensätze mit der Kardinalität + oder \* (Attribut `fkBogenZahl` Tabelle `Bogen`) muss die Eindeutigkeit des Feldes `fkEindeutigBogenFeld` gewährleistet sein (Fehlerart TDS in Kapitel 2.6.2).
- Für jeden Teildatensatz mit mehrfach anlegbaren Mutterteildatensätzen ist zu prüfen, ob ein Mutterteildatensatz existiert, für den die Inhalte des Attributes `IdBogenFeldMutter` und des identifizierenden Attributes des Mutterteildatensatzes (definiert durch `fkEindeutigBogenFeld` in der Tabelle `Bogen`) gleich sind (Kapitel Fehlerart TDS in Kapitel 2.6.2).

#### *Prüfungen auf doppelte Datensätze*

Ein übermittelter Datensatz wird von der datenentgegennehmenden Stelle nicht angenommen, wenn bereits vorher ein anderer Datensatz mit derselben Registriernummer, Vorgangsnummer *und* Versionsnummer übermittelt worden ist.

Die *Wertebereichsüberprüfungen einschließlich Muss-/Kann-Prüfung* und die *Plausibilitätsprüfungen* werden für jeden übermittelten Datensatz durchgeführt.

## 2.5.2 Stornierung von Datensätzen

Datensätze können storniert werden, damit der Vollständigkeitsabgleich der gelieferten Datensätze korrekt geführt werden kann.

Dazu ist das Zusatzfeld `storno = 1` zu setzen. Die datenentgegennehmende Stelle wird dadurch veranlasst, den betreffenden Datensatz einschließlich aller Vorversionen und Teildatensätze als „storniert“ zu kennzeichnen. Der Stornovorgang wird in der Datenbestätigung protokolliert.

### Achtung:

Der mit dem Storno-Zusatz gekennzeichnete Datensatz muss ebenfalls eine um eins erhöhte Versionsnummer enthalten, um die Stornierung unabhängig von der Reihenfolge der Verarbeitung von Datensätzen sicherzustellen. Ein Storno mit einer bereits verwendeten Versionsnummer wird zurückgewiesen (Bestätigungsstatus `FEHLER`, Fehlerart `DOPPELT`). Ein Stornoversuch eines noch nicht übermittelten Datensatzes wird ebenfalls zurückgewiesen.

## 2.6 Antwortdateien

Für jede empfangene Transaktionsdatei generiert die Datenstelle automatisch eine Antwortdatei (ZIP-Archiv), welches nach folgendem Schema benannt ist:

A-<KodeDatenstelle>-<Registrierkode>-<Transaktionsnummer>.ZIP
---

### Beispiel:

A-KR05-001-777.ZIP

Das ZIP-Archiv enthält die Bestätigungsdatei (2.6.1) und die Fehlerdatei (2.6.2), welche Informationen über die Prüfergebnisse der übermittelten Datensätze enthalten. Beide Dateien sind so aufgebaut, dass sie von der Erfassungssoftware maschinell weiterverarbeitet werden können.

## 2.6.1 Die Bestätigungsdatei

Die Bestätigungsdatei ist eine CSV-Datei, welche für jeden übermittelten Datensatz eine Zeile mit dem Prüfergebnis enthält. Sie ist nach folgendem Schema benannt:

```
B-<KodeDatenstelle>-<Registrierkode>-<Transaktionsnummer>.txt
```

### Beispiel:

B-KR05-DS001-0777.txt ist der Name einer Bestätigungsdatei mit folgenden Bestandteilen:

- KR05: Kode der datenentgegennehmende Stelle (Epidemiologisches Krebsregister NRW)
- DS001: Registrierungskode des empfangenden Softwaresystems
- 777: Transaktionsnummer der Datenlieferung, auf welche sich die Antwortdatei bezieht.

Der einzelne Datensatz kann über die Kombination von Registriernummer, Vorgangsnummer und Versionsnummer eindeutig reidentifiziert werden. Im Fehlerfall (*Bestätigungsstatus FEHLER*) werden die Fehler in einer separaten Datei (Kapitel 2.6.2) aufgelistet:

```
<RegistrierNr>;<VorgangsNr>;<VersionNr>;<Modul>;<SPEZVERSION>;<BestätigungsStatus><CR><LF>
```

mit:	
{ }	Wiederholung
[ ]	Option
<CR>	Carriage Return (ASCII 13)
<LF>	Line Feed (ASCII 10)
<BestätigungsStatus>	Werte sind OK oder FEHLER
<RegistrierNr>	Registriernummer des Dokumentationssystem
<VorgangsNr>	Vorgangsnummer
<VersionNr>	
<Modul>	Name des übermittelten Moduls (Attribut Modul.name) <sup>12</sup>
<SPEZVERSION>	Version der gültigen Spezifikation (siehe Attribut Version.name)

### Beispiel:

---

```
KR05-DS001;261012305;1;GEKID;GEKID-2009;FEHLER<CR><LF>
KR05-DS001;261012306;1;GEKID; GEKID-2009;OK<CR><LF>
KR05-DS001;261012307;2;GEKID; GEKID-2009;OK<CR><LF>
KR05-DS001;261012308;1;GEKID; GEKID-2009;OK<CR><LF>
KR05-DS001;261012310;1;GEKID; GEKID-2009;FEHLER<CR><LF>
..
```

## 2.6.2 Die Fehlerdatei

Die Erfassungssoftware muss in der Lage sein, die Fehlermeldungen und Warnungen der datenentgegennehmenden Stelle einzulesen und den Anwender durch eine möglichst komfortable Nachbearbeitungsfunktion für die betreffenden Datensätze zu unterstützen.

Die Fehlerdatei hat einen Namen nach folgendem Schema:

```
F-<KodeDatenstelle>-<Registrierkode>-<Transaktionsnummer>.txt
```

### Beispiel:

```
F-KR05-DS001-0777.txt
```

Voraussetzung für diese Funktionalität ist, dass von der datenentgegennehmenden Stelle die Fehlermeldungen in einer Fehlerdatei übermittelt werden.

Die Zeilen der Fehlerdatei sind folgendermaßen aufgebaut:

```
<Re-
gistrrierNr>;[<Modul>];[<VorgangsNr>];[<VersionNr>];[<SPEZVERSION>];<Fehlerart>;[<Regelnr>
];[<Regeltyp>];[<Liste>];[<Meldung><CR><LF>
```

mit:

{ }	Wiederholung
[ ]	Option
<CR>	Carriage Return (ASCII 13)
<LF>	Line Feed (ASCII 10)
<Modul>	Modul
<RegistrierNr>	Registriernummer des Dokumentationssystem
<VorgangsNr>	Vorgangsnummer
<VersionNr>	
<SPEZVERSION>	Version des Spezifikation
<Fehlerart>	STEUER = Formatfehler der Steuerdatei EXPORT = Formatfehler der Exportdatei

	DOPPELT = bereits vorhandener Datensatz wird erneut übermittelt
	TDS = Vollständigkeit und Version der Teildatensätze
	WERT = Wertebereichsverletzung
	REGEL = Plausibilitätsverletzung
<Regelnr>	Nummer der Regel (idRegeln in Tabelle Regeln), nur wenn Fehlerart=REGEL vergeben, ansonsten leer
<Regeltyp>	nimmt die Werte H (=hart) oder D (=Warnung)
<Liste>	In Abhängigkeit von der Fehlerart entweder Liste von Teildatensätzen oder von Bogenfeldern
<Meldung>	Fehlermeldung als Freitext, z.Z. nicht fest vorgegeben

Für jeden gefundenen Fehler wird eine eigene Zeile angehängt!

Die Fehlerarten beziehen sich auf die in der BQS-Spezifikation für QS-Dokumentation definierten Prüfungen. Es gibt für jeden Fehler auszufüllende Spalten bzw. Felder (<Registriernr> und <Fehlerart>). Andere Spalten bleiben bei einzelnen Fehlerarten leer. Z.B. braucht bei einem Fehler der Steuerdatei nur die Fehlermeldung übermittelt werden. Tabelle 3 gibt einen Überblick darüber, unter welchen Bedingungen in den Feldern der Fehlerdatei Angaben erforderlich sind.

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

Die *Bogenfeldliste* umfasst ein oder mehrere Namen von Bogenfeldern, welche einen Bezug zum Fehler haben. Bei der Fehlerart WERT enthält die Liste nur ein Element. Der Bogenfeldname umfasst auch den Namen des zugehörigen Teildatensatzes<sup>13</sup> (Beispiel: TUBASIS[2].TULOK).

Tabelle 3: Ausfüllen der Felder der Fehlerdatei in Abhängigkeit von den Fehlerarten (Minuszeichen bedeutet "kein Wert" bzw. Nicht-Ausfüllen)

Fehlerart	<Modul>	<Vorgangsnr>/<VersionNr>	<SPEZVERSION>	<Regelnr>	<Regeltyp>	<Liste>	<Meldung>
STEUER	-	-	-	-	-	-	ja
EXPORT	ja	-	ja	-	-	<Bogen>	ja

<sup>13</sup> Der Bezug zum Modul kann entfallen, da es über die Vorgangsnummer identifiziert werden kann.

Fehlerart	<Modul>	<Vorgangsnr>/<VersionNr>	<SPEZVERSION>	<Regelnr>	<Regeltyp>	<Liste>	<Meldung>
DOPPELT	ja	ja	ja	-	-	-	ja
TDS	ja	ja	ja	-	-	<Bogenliste>	ja
WERT	ja	ja	ja	-	-	<Bogenfeldliste> <sup>14</sup>	ja
REGEL	ja	ja	ja	ja	ja	<Bogenliste>	ja

#### Standardisierung der Meldungen bei Bestätigungsstatus mit Fehlerart *DOPPELT*

Es wurde bereits ein anderer Datensatz mit derselben Registriernummer und Versionsnummer übermittelt.

#### Standardisierung der Meldungen bei der Fehlerart *WERT*

Bei feldbezogenen Fehlern sind die standardisierten Fehlermeldungen der BQS-Spezifikation für QS-Dokumentationssoftware zu verwenden.

#### Standardisierung der Meldungen bei Bestätigungsstatus mit Fehlerart *TDS*

- Wenn ein obligatorischer Teildatensatz (Attribut `Bogen.fkBogenZahl` ist + oder 1, oder ein zu einem Kindteildatensatz zugehöriger Mutterteildatensatz) eines Vorganges in den Exportdateien einer Transaktion nicht vorkommt, so ist folgende Fehlermeldung auszugeben:

Erforderlicher Teildatensatz `<Bogen.name>` ("`<Bogen.bezeichnung>`") existiert nicht.

- Wenn die Existenzbedingung eines Kindteildatensatzes im zugehörigen Mutterteildatensatz erfüllt ist, aber kein Kindteildatensatz vorhanden ist, so ist folgende Fehlermeldung auszugeben:

Die Angaben im Datensatz erfordern einen Teildatensatz `<Bogen.name>` ("`<Bogen.bezeichnung>`"). Dieser fehlt.

<sup>14</sup> In der Regel wird hier nur ein Bogenfeld aufgeführt. Ausnahme ist, wenn Kombinationsfelder geprüft werden: ENTLDIAG|ENTLDIAGVERS u. a.

- Wenn die Existenzbedingung eines Kindteildatensatzes im zugehörigen Mutterteildatensatz nicht erfüllt ist, aber trotzdem ein Kindteildatensatz existiert, so ist folgende Fehlermeldung zu erzeugen:

*Die Angaben im Datensatz lassen keinen weiteren Teildatensatz <Bogen.name> ("*<Bogen.bezeichnung>*") zu, obwohl ein solcher übermittelt wurde.*

## **A Referenzen**

[ONKOWL2008]:

Spezifikation für Tumordokumentation mit technischer Dokumentation.

30.06.2007

Düsseldorf: BQS Bundesgeschäftsstelle Qualitätssicherung gGmbH

Online verfügbar unter: