

Einführung der Gesundheitskarte

Speicherstrukturen der eGK für Gesundheits- anwendungen

Version: 1.6.0
Stand: 18.03.2008
Status: freigegeben

Dokumentinformationen

Änderungen zur Version 1.4.0

Die Umsetzung des Löschens von eVerordnungen durch den ERASE-Befehl wurde aufgenommen.

Über ein Testkennzeichen soll die Teilnahme an den Testverfahren der eGK ausgewertet werden können (Kap. 10)

Anhang A (Netto-Speicherplatzbedarf) wurde angepasst: Vorkehrungen für die Organspendeerklärung (als Teil der Notfalldaten) wurden berücksichtigt.

Im gesamten Dokument wurden die Container-Bezeichnungen geändert: EF.eVerordnungstickets statt EF.eRezept_Ticket, EF.StatusVerordnungen statt EF.StatusRezept, EF.eVerordnungsContainer statt EF.eRezept_Container. Diese Änderungen sind nachfolgend nicht gesondert hervorgehoben.

Inhaltliche Änderungen gegenüber der letzten freigegebenen Version sind gelb markiert. Sofern ganze Kapitel eingefügt wurden, wurde zur besseren Lesbarkeit lediglich die Überschrift durch gelbe Markierung hervorgehoben.

Referenzierung

Das Dokument wird von anderen gematik-Dokumenten referenziert als:

[gemeGK_Fach]	gematik (18.03.2008): Einführung der eGK - Speicherstrukturen der eGK für Gesundheitsanwendungen Version 1.6.0
---------------	--

Dokumentenhistorie

Version	Stand	Kap.	Grund der Änderung, besondere Hinweise	Bearbeitung
0.0.1	08.02.07		Neuerstellung mit dem Ziel der Vereinheitlichung der Notation der tabellarischen Strukturdefinitionen und der darin verwendeten Datentypen sowie die Vereinheitlichung der Versionsnummern und Timestamps.	Gematik, AG1
0.0.2	15.02.07		Einarbeitung interner Kommentare	gematik, AG1
0.0.3	21.02.07		Version zur Übergabe an QS	gematik, AG1
0.0.4	27.02.07		Einarbeitung der Kommentare	gematik, AG1
0.9.0	02.03.07		zur Freigabe vorgelegt	gematik
1.0.0	02.03.07		freigegeben	gematik
1.0.1	03.04.07	A.1	Dokument Speicherstrukturen der eGK v1.1.1	gematik, AG1

Version	Stand	Kap.	Grund der Änderung, besondere Hinweise	Bearbeitung
			eingefügt	
1.1.0	04.05.07		nicht freigegeben	gematik, AG1
1.1.2	10.05.07		Änderungen zur Verwaltung freiwilliger Anwendungen. Klarstellung des Initialwertes in StatusVSD. Ergänzung der Datei EF.Version Redaktionelle Überabreitung	gematik, AG1
1.2.0	15.05.07		freigegeben	gematik
1.2.1	08.08.07		Einarbeitung SRQ 581	gematik, AG1
1.3.0	24.08.07		freigegeben	gematik
1.3.4	21.11.07		Ersetzung des Begriffs OID durch ObjectID Initialisierung des Feldes Version in der Datei EF.StatusNotfalldaten geändert. StatusPIN entfernt.	SPE/FA
1.4.0	18.12.07		freigegeben	gematik
1.4.1	27.02.08		Container umbenannt	SPE/FA
1.5.0	29.02.08		freigegeben	gematik
1.5.1	18.03.08	4 10	Änderungen beim Netto-Speicherplatzbedarf ERASE-Befehl aufgenommen Einfügen Testkennzeichen	SPE/FA
1.6.0	18.03.08		freigegeben	gematik

Inhaltsverzeichnis

Dokumentinformationen	2
Inhaltsverzeichnis.....	4
1 Zusammenfassung	6
2 Einführung.....	7
2.1 Zielsetzung und Einordnung des Dokuments	7
2.2 Zielgruppe	7
2.3 Geltungsbereich	7
2.4 Arbeitsgrundlagen.....	7
2.5 Abgrenzung des Dokuments	8
2.6 Methodik und Notation.....	8
3 Versichertendaten.....	9
3.1 Einleitung	9
3.2 Allgemeine Versicherungsdaten (EF.VD).....	9
3.3 Geschützte Versichertendaten (EF.GVD).....	10
3.4 Persönliche Versichertendaten (EF.PD).....	11
3.5 VSD Status (EF.StatusVD).....	11
4 eVerordnung.....	13
4.1 Einleitung	13
4.2 Ablaufbeschreibung.....	17
4.2.1 eVerordnung auf der eGK speichern	17
4.2.2 eVerordnung von der eGK dispensieren.....	18
4.2.3 eVerordnung auf der eGK löschen	19
4.3 eTickets (EF.eVerordnungsTickets).....	20
4.4 eVerordnung (EF.eVerordnungsContainer).....	21
4.5 eVerordnung Status (EF.StatusVerordnungen).....	22
5 Notfalldaten	23
5.1 Einleitung	23
5.2 Notfalldaten (EF.eNotfalldaten)	23
5.3 Notfalldaten Status (EF.StatusNotfalldaten).....	24

6	Protokolldaten	25
6.1	Protokolldaten (EF.Logging).....	25
7	Dienstverweis	27
7.1	Dienstverweis (EF.Verweis).....	27
8	Einwilligung	28
8.1	Einleitung	28
8.2	Einwilligung (EF.Einwilligung).....	29
9	Version	30
9.1	Einleitung	30
9.2	Version (EF.Version)	30
10	Testkennzeichen	31
10.1	Einleitung	31
10.2	Testkennzeichen (EF.TTN).....	31
Anhang A Netto-Speicherplatzbedarf der eGK		32
Anhang B		34
B1	Abkürzungen.....	34
B2	Glossar	35
B3	Abbildungsverzeichnis	35
B4	Tabellenverzeichnis.....	35
B5	Referenzierte Dokumente.....	36

1 Zusammenfassung

Das vorliegende Dokument „Speicherstrukturen der eGK für Gesundheitsanwendungen“ ist Teil der Arbeitsergebnisse der gematik, die auf Weisung des BMG im Rahmen der „Verordnung über Testmaßnahmen für die Einführung der elektronischen Gesundheitskarte“ vom 05.10.2006 [RVO2006] entstehen.

Das Dokument fasst die Daten und Datenstrukturen zusammen, die für die Realisierung der Anwendungen für Versichertendatenmanagement, Notfalldatenmanagement, Verwaltungsdatenmanagement, Verwaltung freiwilliger Anwendungen und Protokollierung maßgeblich sind.

Zur Vermeidung von Redundanzen wird dieses Dokument durch die Facharchitekturen und weitere Spezifikationen referenziert, soweit die Beschreibung und Festlegung der Fachdaten behandelt wird.

2 Einführung

2.1 Zielsetzung und Einordnung des Dokuments

Das vorliegende Dokument beinhaltet die Spezifikation der Speicherstrukturen der Fachanwendungen auf der eGK im Rahmen der Anwendungen des § 291 a SGB V sowie der Verordnung über Testmaßnahmen zur Einführung der elektronischen Gesundheitskarte. Das Dokument bildet damit zunächst die Grundlage für die Realisierung und Test dieser Anwendungen.

2.2 Zielgruppe

Das vorliegende Dokument richtet sich an Kartenhersteller und Entwickler von Kartenanwendungen und Konnektoren.

Es ist gleichzeitig Grundlage für weitere Spezifikation der gematik im Rahmen der Einführung der Gesundheitskarte.

Begrifflichkeiten aus objektorientierten Entwicklungsansätzen und der UML-Modellierung werden als bekannt vorausgesetzt.

2.3 Geltungsbereich

Das Dokument gilt für alle Personen, Institutionen und Systeme, die an der Umsetzung der „Verordnung über Testmaßnahmen für die Einführung der elektronischen Gesundheitskarte“ [RVO2006] in der Bundesrepublik Deutschland beteiligt sind.

In dieser Version gilt das Dokument für die Funktionsabschnitte 1-4, die sich auf die in der Rechtsverordnung beschriebenen Abschnitte beziehen.

2.4 Arbeitsgrundlagen

Bei der Erarbeitung dieser Spezifikation wurden folgende Arbeitsgrundlagen berücksichtigt:

Tabelle 1: Fachkonzepte (FK) und Facharchitekturen (FA) zur Einführung der Gesundheitskarte

Fachanwendung	Inhalt	Querverweis
Versichertenstammdatenmanagement (VSDM)	Verwaltung und Prüfung der Versichertenstammdaten auf der eGK	[gemFK_VSDM] [gemFA_VSDM]

Fachanwendung	Inhalt	Querverweis
Verordnungsdatenmanagement (VODM)	Management von eVerordnungen für Versicherte	[gemFK_VODM] [gemFA_VODM]
Daten für die Notfallversorgung (NFDM)	Management von Notfalldaten der Versicherten	[gemFK_NFDM] [gemFA_NFDM]
Kartenmanagement der eGK	Spezifikation von Schnittstellen von Karten- und Kartenanwendungsmanagementsystemen zur Telematikinfrastruktur	[gemFK_CMSeGK] [gemFA_CMSeGK]
Verwaltung freiwilliger Anwendungen (VFA)	Verwaltung von Einwilligungen zu freiwilligen Anwendungen	[gemFK_VFA] [gemFA_VFA]
Anwendungen des Versicherten (ADV)	Verwaltung von Anwendungen und Anwendungsdaten durch den Versicherten	[gemFK_ADV] [gemFA_ADV]

2.5 Abgrenzung des Dokuments

Es handelt sich um eine technisch motivierte Beschreibung. Die fachliche Begründung und Erläuterung ist Teil der in Tabelle 1 benannten Fachkonzepte.

Die technische Abbildung der fachlogischen Prozesse und Informationsobjekte auf Dienste, technische Komponenten und Datenmodelle ist Gegenstand der separat beschriebenen Facharchitekturen.

2.6 Methodik und Notation

Das Dokument stellt die Speicherstrukturen der eGK gegliedert nach Fachanwendungen tabellarisch dar. Die verwendeten Datentypen sind in Tabelle 2 definiert. Längenangaben für Informationselemente erfolgen in Byte. Hexadezimale Werte werden mit dem Präfix „0x“ gekennzeichnet. Werte ohne Präfix sind dezimal.

Tabelle 2 - Definition der Datentypen

Datentyp	Definition
ALPHA	Text String nach ISO8859-15. NULL (0x00) terminiert, falls die Textlänge die Größe des Informationselements unterschreitet.
BCD	„Binary Coded Decimal“, z.B. 0x20 0x07 für “2007”
BINÄR	vorzeichenloser, ganzzahliger, numerischer Wert in binärer Big-Endian-Darstellung. Beispielhaft sei hier noch erwähnt, dass der Wertebereich eines BINÄR-Wertes mit Länge 1 dementsprechend 0..255 ist und mit der Länge 2 0..65535

3 Versichertendaten

3.1 Einleitung

Die Versichertendaten werden auf der eGK in vier Dateien abgelegt:

- Allgemeine Versicherungsdaten (EF.VD)
- Geschützte Versichertendaten (EF.GVD)
- Persönliche Versichertendaten (EF.PD)
- VSD Status (EF.StatusVD)

Die Datei EF.StatusVD wird genutzt, um das Datum der letzten Aktualisierung, die Version und einen Status zu sichern. Der Status zeigt an, ob der letzte Aktualisierungsprozess vollständig durchlaufen ist und die Daten über die übrigen drei VSD-Dateien hinweg technisch konsistent sind. Schreibabbrüche während der Aktualisierung lassen sich somit eindeutig erkennen.

Es sei ausdrücklich auf den Beschluss des BMG für Release 0 verwiesen, in dem für eine Übergangsphase abweichende Definitionen zu den Inhalten der einzelnen Dateien vorgegeben sind.

3.2 Allgemeine Versicherungsdaten (EF.VD)

Der Bereich der allgemeinen Versicherungsdaten enthält Vertragsdaten zum Versicherungsverhältnis.

Zur Unterstützung der Szenarien für Release 0 und Release 1 wird auf Beschluss des BMG zunächst in EF.VD neben den allgemeinen Versicherungsdaten zusätzlich eine Kopie der geschützten Versichertendaten EF.GVD abgelegt.

Tabelle 3 - Struktur der Datei EF.VD

Informationselement	Länge in Byte	Typ	Initialwert	Bemerkung
Offset Start VD	2	BINÄR		Siehe 1.
Offset Ende VD	2	BINÄR		Siehe 1.
Offset Start GVD	2	BINÄR		Siehe 2.
Offset Ende GVD	2	BINÄR		Siehe 2.
VD	variabel			Siehe 3., siehe 5
GVD	variabel			Siehe 4., siehe 5

1. Der Offset berechnet sich ab Start der Datei und wird in Byte angegeben. Durch die Länge des Offset-Bereiches ist der kleinstmögliche Offset-Wert 8.
2. Liegen keine geschützten Versichertendaten GVD vor, zum Beispiel im Falle einer privaten Krankenversicherung, so werden die Felder Offset Start GVD und Offset Ende GVD jeweils mit hexadezimal ‚FFFF‘ belegt um das Feld GVD als leer zu markieren. Der Offset berechnet sich ab Start der Datei und wird in Byte angegeben. Durch die Länge des Offset-Bereiches ist der kleinst mögliche Offset Wert 8.
3. Die allgemeinen Versicherungsdaten VD selbst werden als XML-Daten gemäß vorgegebenem XML-Schema, gzip-komprimiert und nicht verschlüsselt innerhalb der Datei abgelegt. Der zu verwendende Zeichensatz für die fachlichen Inhalte ist ISO8859-15.
4. Die allgemeinen geschützten Versichertendaten GVD selbst werden als XML-Daten gemäß vorgegebenem XML-Schema, gzip-komprimiert und nicht verschlüsselt innerhalb der Datei abgelegt. Der zu verwendende Zeichensatz für die fachlichen Inhalte ist ISO8859-15.
5. Die Reihenfolge der Felder VD und GVD ist durch die entsprechenden Offsets definiert und nur exemplarisch wie oben abgebildet. Die maximale Länge ergibt sich aus der in [gemSpec_eGK_P2] definierten Dateigröße für Versicherungsdaten minus dem Speicherplatzbedarf für den Offsetbereich (8 Byte).

3.3 Geschützte Versichertendaten (EF.GVD)

Die geschützten Versichertendaten enthalten Informationen über den Teil der Versichertendaten, die besonders schützenswert sind. Der Zugriff auf diese Daten ist nur nach Freischaltung dieser Datei durch eine berechnigte Karte möglich.

Es wird hier ausdrücklich darauf hingewiesen, dass auch für Release 0 und Release 1 die geschützten Versichertendaten in der Datei EF.GVD abgelegt werden. Zusätzlich wird eine Kopie dieser Daten in der Datei EF.VD abgelegt.

Tabelle 4 - Struktur der Datei EF.GVD

Informationselement	Länge in Byte	Typ	Initialwert	Bemerkung
Länge GVD	2	BINÄR	0	Siehe 1.
GVD	variabel			Siehe 2.
<ol style="list-style-type: none"> Liegen keine geschützten Versichertendaten GVD vor, zum Beispiel im Falle einer privaten Krankenversicherung, so wird das Feld Länge mit 0 belegt um das Feld GVD als leer zu markieren. Ansonsten wird die Länge des Feldes GVD in Byte angegeben. Der Maximalwert ergibt sich aus der in [gemSpec_eGK_P2] definierten Dateigröße für Geschützte Versichertendaten minus dem Speicherplatzbedarf für das Längengeld (2 Byte) Die geschützten Versichertendaten GVD selbst werden als XML-Daten gemäß vorgegebenem XML-Schema, gzip-komprimiert und nicht verschlüsselt innerhalb der Datei abgelegt. Der zu verwendende Zeichensatz für die fachlichen Inhalte ist ISO8859-15. 				

3.4 Persönliche Versichertendaten (EF.PD)

In EF.PD werden die persönlichen Versichertendaten abgelegt.

Tabelle 5 - Struktur der Datei EF.PD

Informationselement	Länge in Byte	Typ	Initialwert	Bemerkung
Länge PD	2	BINÄR		Siehe 1.
PD	variabel			Siehe 2.
<ol style="list-style-type: none"> Gibt die Länge des Feldes PD in Byte an. Die maximale Länge ergibt sich aus der in [gemSpec_eGK_P2] definierten Dateigröße minus des Speicherplatzbedarfes für das Längengeld (2 Byte). Die persönlichen Versichertendaten PD selbst werden als XML-Daten gemäß vorgegebenem XML-Schema, gzip-komprimiert und nicht verschlüsselt innerhalb der Datei abgelegt. Der zu verwendende Zeichensatz für die fachlichen Inhalte ist ISO8859-15. 				

3.5 VSD Status (EF.StatusVD)

Im Container EF.StatusVD werden Statusinformationen, Zeitstempel und Versionsinformationen abgelegt, die die Daten der anderen VSD-Container betreffen (EF.VD, EF.GVD, EF.PD).

Tabelle 6 - Struktur der Datei EF.StatusVD

Informationselement	Länge in Byte	Typ	Initialwert	Bemerkung
Status	1	ALPHA	„0“	„1“ = Transaktionen offen „0“ = keine Transaktionen offen
Timestamp	14	ALPHA	Siehe 1.	Timestamp der letzten Aktualisierung der VSD auf dem VSDD im Format YYYYMMDDhhmmss
Version	5	BCD	0x0000000000	siehe 2.
Reserviert	5	BINÄR	0	
<ol style="list-style-type: none"> Das Informationselement Timestamp wird mit dem Zeitstempel des Personalisierungszeitpunktes vorbelegt Versionsnummer der VSD Struktur im Format XXXYYYZZZZ. Diese wird beim Personalisieren der Karte mit der Versionsnummer der VSD aus der Facharchitektur (siehe [gem-FA_VSDM]) gefüllt. 				

4 eVerordnung

4.1 Einleitung

eVerordnungen werden für den Fall der Offline-Speicherung auf der eGK gzip-komprimiert und verschlüsselt abgelegt, die Schlüssel werden gemeinsam mit einem Anzeigeteil für den Versicherten in so genannten eTickets gespeichert. eVerordnungen werden durch den Arzt **qualifiziert elektronisch** signiert. Erst mit seiner Signatur erhalten eVerordnungen ihre Gültigkeit. Bestandteil der Signatur der eVerordnungen sind Zertifikate vom HBA des Arztes, das QES-Zertifikat und das damit in Zusammenhang stehende Attributzertifikat mit der Rolle Arzt, die im Folgenden stets im Zusammenhang betrachtet werden.

Für die Ablage der eVerordnungen auf der eGK gibt es 3 Dateien:

- EF.eVerordnungTickets mit den Schlüsseln zum Zugriff auf die eVerordnungen (siehe Abschnitt 4.3)
- EF.eVerordnungsContainer mit den eVerordnungen selbst und den dazugehörigen Zertifikaten (siehe Abschnitt 4.4)
- EF.StatusVerordnungen (siehe Abschnitt 4.5)

Die eVerordnungen selbst sind in einem XML-Dokument gespeichert. Die XML-Dokumentstruktur wird durch die gematik in einem entsprechenden XML-Schema vorgegeben.

Da die eVerordnungen selbst unterschiedliche Informationsgehalte und damit auch Längen haben können, ist auch die Größe der auf der eGK abzulegenden Datensätze variabel. Eine Ablage in einem Format mit fixer Länge ist so auf der eGK nicht möglich. Da ein Datensatz einer eVerordnung in jedem Falle länger sein wird als 255 Byte, ist eine Ablage der eVerordnungen als ein Record mit variabler Länge wegen deren Beschränkung auf 255 Byte nicht möglich.

Aus diesem Grunde werden alle eVerordnungen in einer Datei auf der eGK abgelegt. Dieses File ist auf Kartenebene nicht weiter strukturiert. eVerordnungen werden als logisch unabhängige Blöcke in dieses File geschrieben. Die Verwaltung der einzelnen Blöcke im Sinne der Positionierung der eVerordnungen in dem File obliegt der lesenden und schreibenden Anwendung, in diesem Falle dem Konnektor. Dies betrifft Operationen wie Lesen, Schreiben oder Löschen eines Rezeptes.

Der Offset der einzelnen eVerordnungen (Anfang und Ende) wird in einem Pointer-Bereich am Beginn der Datei EF.eVerordnungsContainer gespeichert, wobei die Zuordnung der einzelnen Slots im Pointer-Bereich der Zuordnung der Records in der Datei EF.eVerordnungTickets entspricht (vgl. Abbildung 1).

Das unverschlüsselte Zertifikat des Verordners wird zur Speicheroptimierung getrennt von den eVerordnungen in die Datei EF.eVerordnungsContainer geschrieben. Mehrere während eines Arztbesuchs entstandene eVerordnungen des gleichen Verordners können so mit Verweisen auf nur ein Zertifikat geschrieben werden, was signifikant Speicherplatz

spart (pro weiterem Rezept etwa 3500 Byte). Die technische Umsetzung ist exemplarisch in Abbildung 1 (Ticket A und B) beschrieben.

Die Offsets des Anfangs und Endes des jeweiligen Zertifikats sowie die der eVerordnung werden im gleichen Slot des Pointer-Bereichs abgelegt.

Die Dispensierung einzelner eVerordnungen auf Wunsch des Versicherten wird dadurch nicht behindert.

Im Folgenden ist das Speicherkonzept für die eVerordnungen beispielhaft dargestellt, insbesondere der Zusammenhang zwischen den Dateien EF.eVerordnungTickets und EF.eVerordnungsContainer.

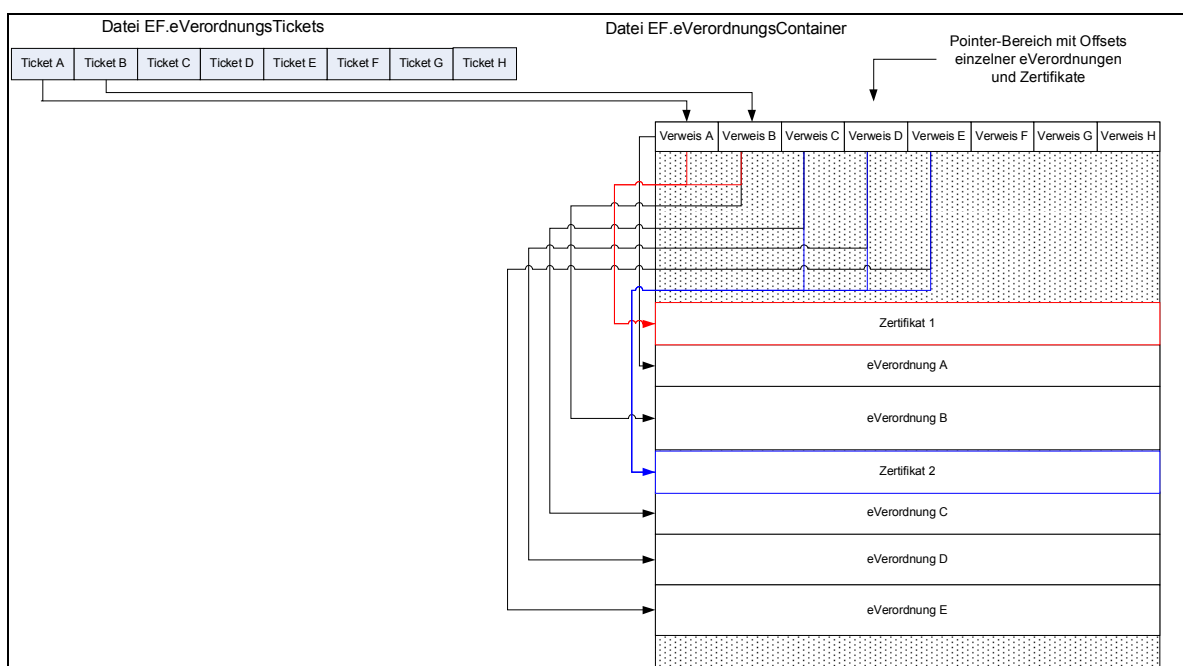


Abbildung 1 Speicherkonzept für eVerordnungen auf der eGK

Entsprechend der Record-Nummer des eTickets (in der Abbildung mit den Buchstaben A-H bezeichnet) wird der im Pointer-Bereich der Datei EF.eVerordnungsContainer gelegene Bereich mit den Offsets für die referenzierte eVerordnung ausgewählt. Dort sind die Beginn- und Ende-Adressen jeweils für die eVerordnung und das Zertifikat gespeichert, die im nachfolgenden Leseprozess für das Einlesen genutzt werden. So referenziert in der Abbildung das eTicket im Record 00 („A“) auf den Speicherbereich „Verweis A“, welcher wiederum auf das Zertifikat „1“ und die eVerordnung „A“ verweist. Das eTicket im Record 01 („B“) entsprechend auf den Speicherbereich „Verweis B“. Dieser verweist auf eVerordnung „B“, die im obigen Beispiel ebenfalls das Zertifikat „1“ verwendet, in diesem Fall also vom gleichen Verordner ausgestellt wurde wie eVerordnung „A“. Die eVerordnungen „C“, „D“ und „E“ verwenden das gemeinsame Zertifikat „2“.

Die hier spezifizierte Speicherstruktur ermöglicht eine Defragmentierung des Inhalts durch den Konnektor zur Rückgewinnung von freien Speicherbereichen, die einzeln zu klein für

die Aufnahme einer eVerordnung bzw. Zertifikats wären. Die Defragmentierung soll spätestens dann automatisch durchgeführt werden, wenn bei einem Arztbesuch noch nicht dispensierte eVerordnungen auf der eGK gespeichert sind und der Speicherplatz für die Aufnahme der neuen Rezepte nicht ausreicht. In der Facharchitektur VODM ist außerdem ein Technischer Use Case definiert (siehe [gemFA_VODM#4.16 TUC_VO_10 eVerordnungs-Container bereinigen auf der eGK], der es einem Primärsystem gestattet, die Defragmentierung explizit zu starten.

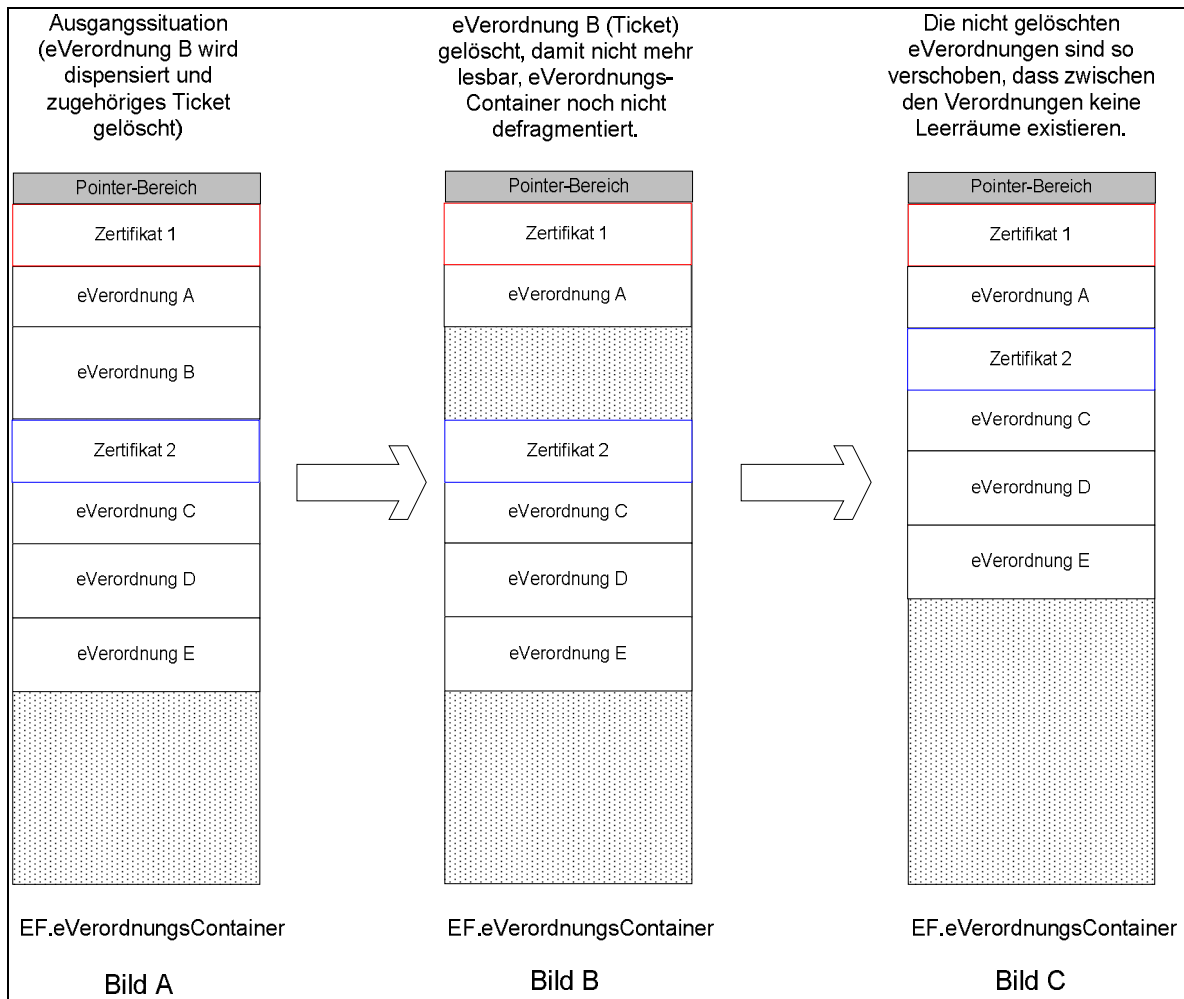


Abbildung 2 Defragmentierung der Datei EF.eVerordnungsContainer

Zum Defragmentieren der Datei EF.eVerordnungsContainer werden die eVerordnungen, durch Auslesen und Schreiben an eine neue Position, verschoben. Danach werden die jeweiligen Offsets im Pointer-Bereich angepasst. Die Integrität der Verordnungen ist bei diesem Vorgang, auch im Falle eines Abbruchs, durch geeignete Maßnahmen sicherzustellen. So muss die eVerordnung dabei jederzeit (also auch bei Abbruch der Defragmentierung durch einen technischen Fehler wie Stromausfall o.ä.) spezifikationskonform gespeichert und lesbar sein. Deshalb muss zum Beispiel das Verschieben von sich überlap-

penden Bereichen (siehe Abbildung 3) unterbleiben, da eine Restaurierung einer durch Überschreiben des Inhalts im Überlappungsbereich zerstörten eVerordnung ohne Entschlüsselung nicht möglich ist. Ist die Datenintegrität während der Defragmentierung nicht sichergestellt, darf nicht defragmentiert werden.

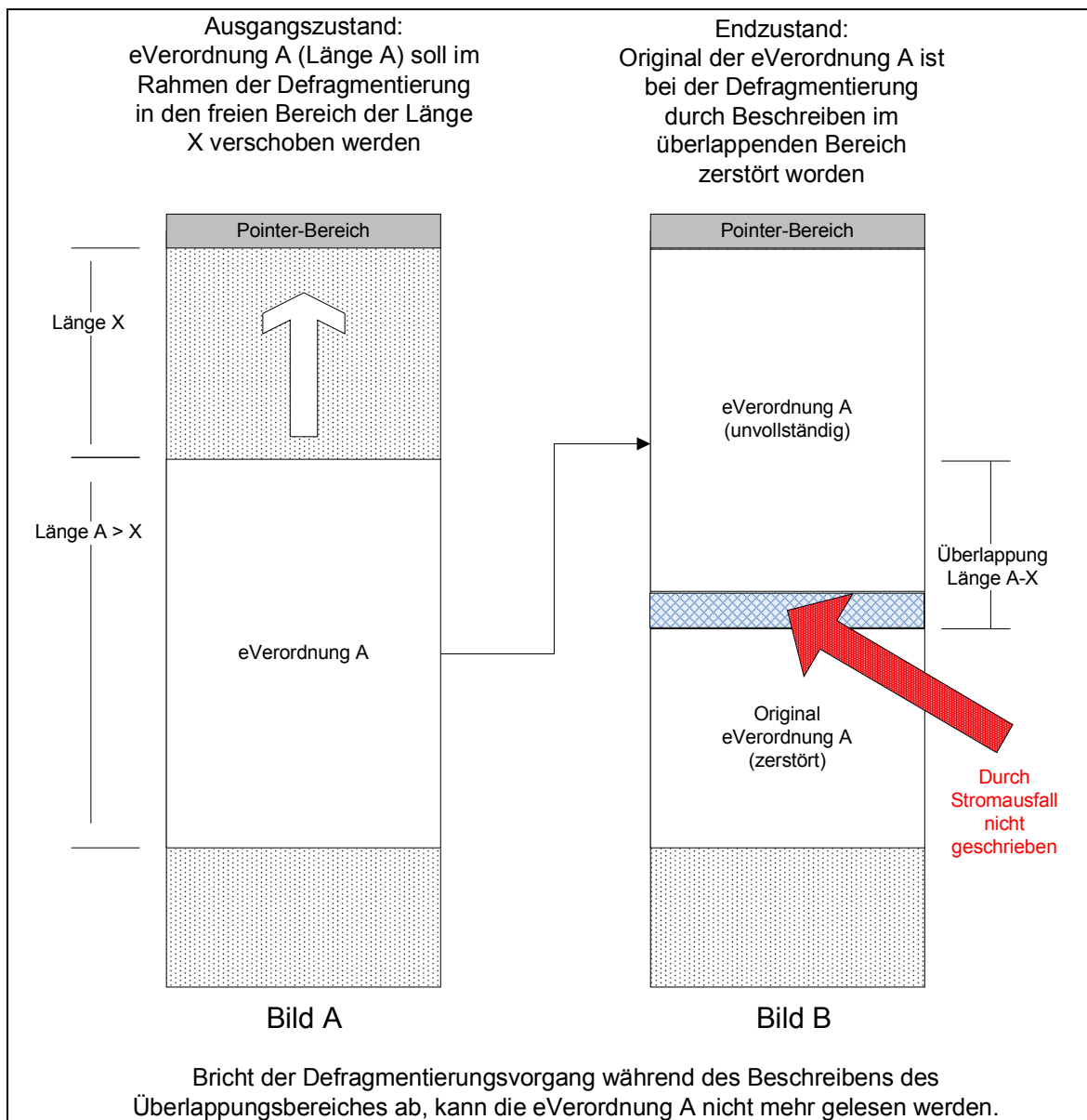


Abbildung 3 Verhinderung von Integritätsproblemen bei Defragmentierung

4.2 Ablaufbeschreibung

4.2.1 eVerordnung auf der eGK speichern

Vor dem Schreiben einer eVerordnung wird ein freier Record in der Datei EF.eVerordnungstickets gesucht. Freie Records zeichnen sich durch eine Belegung aller Felder mit 0x00 aus¹, so z.B. auch das Feld Ausstellungsdatum (0x00 0x00 0x00 0x00) (siehe Tabelle 7). Die Anzahl dort zu speichernder eTickets limitiert die Anzahl auf der eGK speicherbarer eVerordnungen entsprechend der eGK-Spezifikation [gemSpec_eGK_P2] auf 8.

Vor dem Schreiben der eVerordnung in die Datei EF.eVerordnungsContainer wird geprüft, ob dort genügend Speicherplatz vorhanden ist. Im einfachsten Fall sind keine eVerordnungen auf der eGK gespeichert, sodass der gesamte Speicherplatz der Datei EF.eVerordnungsContainer hinter dem Pointer-Bereich frei ist. Freier Speicherplatz kann ansonsten sowohl zwischen zwei eVerordnungen, als auch zwischen der letzten eVerordnung und dem Ende der Datei existieren. Zum Ermitteln der Speicherbelegung muss der Pointer-Bereich in der Datei EF.eVerordnungsContainer gelesen werden. Dabei muss beachtet werden, dass neben der eigentlichen eVerordnung auch das aus der Signatur extrahierte Zertifikat des Verordners mit geschrieben werden muss (erst das QES-Zertifikat, danach das Attributzertifikat binär direkt aufeinander folgend), so dass die Platzprüfung für zwei Bereiche erfolgen muss.

Bei der gleichzeitigen Verordnung von mehreren Rezepten durch einen Verordner braucht zur Speicherplatzersparnis nur ein Zertifikat geschrieben werden. Da die funktionale Schnittstelle zum Konnektor derzeit nur das Schreiben jeweils einer eVerordnung erlaubt, muss der Konnektor die Zertifikatreferenz beim ersten Schreibvorgang auf eine eGK in einer Ablage zwischenspeichern, um es bei weiteren Schreibvorgängen auf die gleiche eGK wieder verwenden zu können. Beim Ziehen und Stecken der Karte sowie beim Einschalten des Lesegerätes muss die Zertifikatreferenz-Ablage gelöscht werden.

¹ Hinweis zur Implementierung: Eine Liste freier Records ist deshalb auch durch den Befehl SEARCH RECORD (siehe [gemSpec_eGK_P1]) mit dem Muster 0x00 in der Länge des Records lt. Tabelle 7 (165 Byte) zu erhalten.

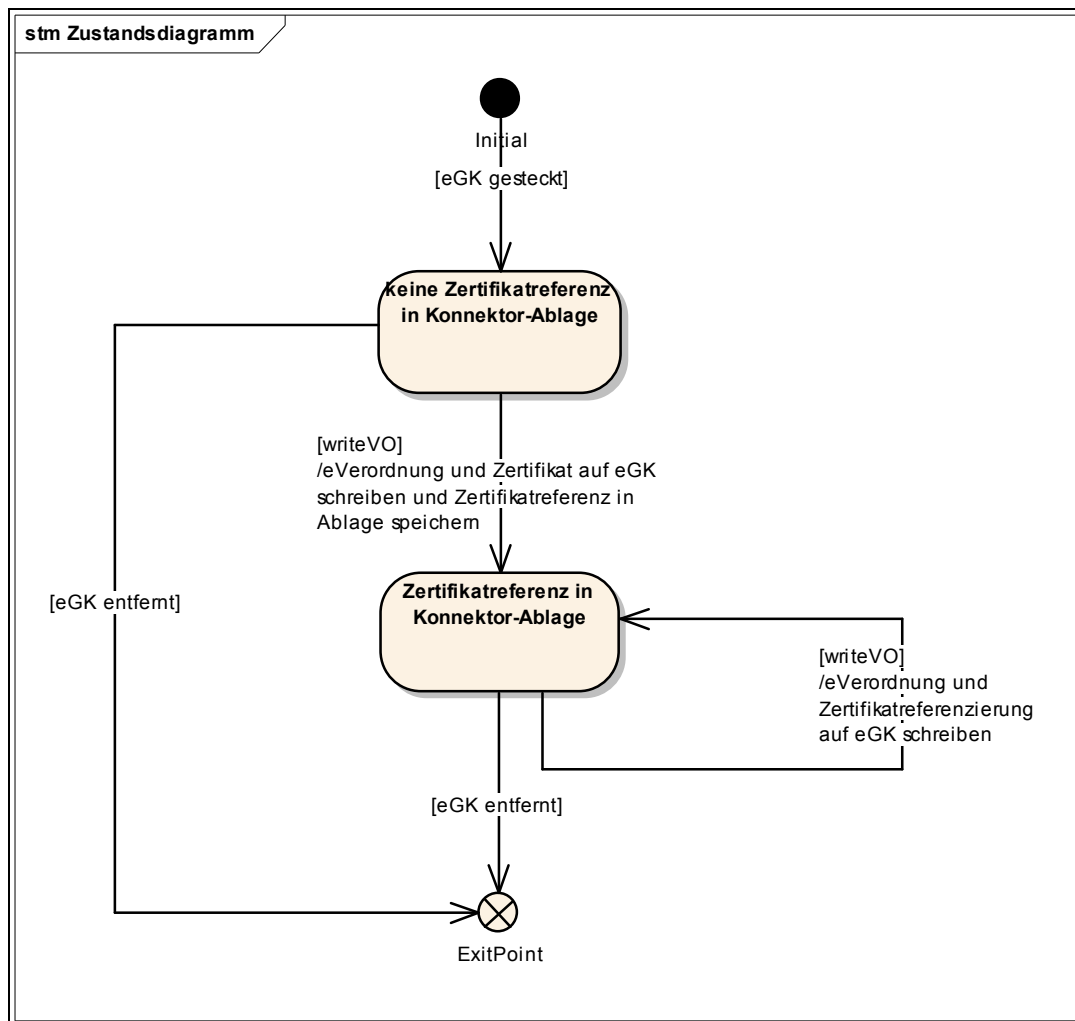


Abbildung 4 Zustandsdiagramm Zertifikatsreferenz im Konnektor

Wird ein zusammenhängender Speicherbereich gefunden, der die eVerordnung aufnehmen kann, so wird die eVerordnung dort gespeichert.

Ist der freie Speicherbereich zu klein für die Aufnahme der neuen eVerordnung und schafft die Defragmentierung der Datei hierbei keine Abhilfe, so erfolgt keine Speicherung auf der eGK und eine entsprechende Fehlermeldung wird an das PVS ausgegeben.

Die Speicherung der eVerordnung erfolgt verschlüsselt, der dazu notwendige vom Konnektor generierte symmetrische Key wird im eTicket gespeichert.

4.2.2 eVerordnung von der eGK dispensieren

Es werden die über eTickets referenzierten Verordnungen von der Karte gelesen, im Konnektor entschlüsselt und im Primärsystem, z.B. des Apothekers, angezeigt. Nach der erfolgreichen Prüfung der Dispensierbarkeit können die eVerordnungen dispensiert werden. Dabei werden neben der eVerordnung die Signatur des Verordners und das zugehörige Zertifikat an das Primärsystem geliefert. Nach der Dispensierung erfolgt das Löschen

des jeweiligen eTickets durch **einen ERASE-Befehl** und somit ein logisches Löschen der eVerordnung. Ein physikalisches Löschen der Verordnung in der Datei EF.eVerordnungsContainer, z. B. durch Löschen der Offsets, ist nicht nötig und erfolgt nicht, da eine Entschlüsselung durch das Löschen des symmetrischen Keys im eTicket nicht mehr möglich ist. Damit wird für das Dispensieren kein Schreibrecht in der Datei EF.eVerordnungsContainer benötigt.

Für das Transaktionshandling, also das quasi gleichzeitige Versenden der Rückmeldung, bestehend aus eVerordnung und Signatur an das Primärsystem, und das Löschen der eVerordnung auf der Karte, wird **das** Flag **eTicket.DelMark** genutzt, welches eine zu dispensierende eVerordnung markiert und vor dem Versenden der Signatur im betreffenden eTicket gesetzt wird. **Das Löschkennzeichen (eTicket.DelMark) hat bei gesetztem DelMark den Wert „1“.** Im Feld eTicket.Heilberufler wird beim Setzen von eTicket.DelMark im Feld eTicket.Heilberufler die Certificate Holder Reference (CHR) des zum Öffnen der eGK benutzten CV-Zertifikats (enthält die ICCSN der entsprechenden Smardcard) in String-Form eingetragen. Im Falle eines Scheiterns des Löschens der eVerordnung auf der eGK kann über den Abgleich der beteiligten Smardcards (nur der gleichen Smardcard ist ein erneuter Zugriff erlaubt) ein doppeltes Einlösen einer eVerordnung **verhindert werden**.

4.2.3 eVerordnung auf der eGK löschen

eVerordnungen können auch ohne gleichzeitiges Dispensieren gelöscht werden. eVerordnungen werden gelöscht, indem das entsprechende eTicket mit dem darin enthaltenen symmetrischen Key durch **einen ERASE-Befehl** gelöscht wird. Die eVerordnung bleibt hierbei unverändert auf der eGK, kann aber, da der symmetrische Key bereits gelöscht wurde, nicht mehr entschlüsselt werden **und ist damit logisch gelöscht**. Die Offsets der eVerordnung und des zugehörigen Zertifikats verbleiben ebenfalls in der Datei EF.eVerordnungsContainer.

4.3 eTickets (EF.eVerordnungTickets)

Das eTicket hat immer eine feste Länge von 165 Byte. Die Länge der Datei EF.eVerordnungTickets beträgt damit 1320 Byte bei geforderten 8 Verordnungen gemäß eGK-Spezifikation [gemSpec_eGK_P2].

Tabelle 7 - Struktur der Datei EF.eVerordnungTickets

Informationselement	Länge in Byte	Typ	Initialwert	Bemerkung
Ausstellungsdatum	4	BCD	0	Format YYYYMMDD
Medikamentenname	70	ALPHA	0x00	Name des Medikamentes
Heilberufler	34	ALPHA	0x00	Name des ausstellenden Heilberuflers bzw. Certificate Holder Reference (CHR) des zum Öffnen der eGK benutzten CV-Zertifikats (enthält die ICCSN der entsprechenden Smard-card)
Key	32	BINÄR	0	Symmetrischer Schlüssel zur Verschlüsselung des Rezepts
DelMark	1	ALPHA	0x00	Markierung für das Löschen der eVerordnung (wird nur während der Dispensier-Transaktion benutzt). Sie hat bei gesetztem DelMark den Wert „1“.
ObjectID	16	BINÄR	0	ObjectID der eVerordnung
Versionskennung XSD	5	BCD	0	CDM_Version aus den Verordnungsdaten (definiert in eVerordnungPayload.xsd) im Format XXXYYYYZZZ
Verordnungsart	1	BINÄR	0	0 – VOD:Arzneimittelverordnung
Algorithmus ID	1	BINÄR	0	ID des Verschlüsselungsalgorithmus zur Verwendung mit dem symmetrischen Schlüssel 1 – AES-128 2 – AES-256
Reserviert	1	BINÄR	0	
<p>1. Der Anzeigenteil für den Versicherten umfasst die Felder Ausstellungsdatum, Medikamentenname und Heilberufler. Es ist nicht geplant, den Anzeigeteil für den Versicherten in fachlichen Abläufen bei der Dispensierung der eVerordnungen zu verwenden, da dafür die gespeicherten Informationen nicht ausreichen. Er dient ausschließlich der Information des Versicherten.</p>				

4.4 eVerordnung (EF.eVerordnungsContainer)

Die Datei EF.eVerordnungsContainer beinhaltet den gesamten, nicht strukturierten Bereich für die Speicherung der eVerordnungen.

Tabelle 8 - Struktur der Datei EF.eVerordnungsContainer

Informationselement	Länge in Byte	Typ	Initialwert	Bemerkung
Offset Start Rezept A	2	BINÄR	0	Siehe 1.
Offset Ende Rezept A	2	BINÄR	0	
Offset Start Zertifikat A	2	BINÄR	0	
Offset Ende Zertifikat A	2	BINÄR	0	
Offset Start Rezept H	2	BINÄR	0	Siehe 1.
Offset Ende Rezept H	2	BINÄR	0	
Offset Start Zertifikat H	2	BINÄR	0	
Offset Ende Zertifikat H	2	BINÄR	0	
eVerordnungen	variabel	BINÄR	0	Siehe 2.
<p>1. Die Elemente Offset Start Rezept, Offset Ende Rezept, Offset Start Zertifikat und Offset Ende Zertifikat bilden einen Pointer auf eine eVerordnung. Insgesamt umfasst der Pointer-Bereich gemäß eGK-Spezifikation [gemSpec_eGK_P2] 8 Einträge (A bis H), von denen hier nur zwei dargestellt sind. Der Offset berechnet sich ab Start der Datei.</p> <p>2. Im Speicherbereich der Datei EF.eVerordnungsContainer, der dem Pointer-Bereich direkt folgt, liegen die gzip-komprimierten und verschlüsselten eVerordnungen gemäß dem im eTicket mit seiner Versionsnummer identifizierten XML-Schema, sowie die dazugehörigen Zertifikate. Diese werden, wenn gleichzeitig mehrere eVerordnungen geschrieben werden, ohne Lücken aneinandergereiht, jeweils referenziert durch Einträge im Pointer-Bereich. Bei Schreibvorgängen in eine teilbelegte Datei EF.eVerordnungsContainer werden dagegen zwangsläufig Lücken verbleiben, da eVerordnungen keine feste Länge haben.</p>				

4.5 eVerordnung Status (EF.StatusVerordnungen)

Die Datei EF.StatusVerordnungen enthält Statusinformationen für die Dateien EF.eVerordnungTickets und EF.eVerordnungsContainer.

Tabelle 9 - Struktur der Datei EF.StatusVerordnungen

Informationselement	Länge in Byte	Typ	Initialwert	Bemerkung
Reserviert	1	BINÄR	0	
Timestamp	14	ALPHA	Siehe 1.	Timestamp des letzten Schreibvorganges im Format YYYYMMDDhhmmss
Version EF.eVerordnungTickets	5	BCD	0x0020010000	Versionsnummer der Datei EF.eVerordnungTickets im Format XXXYYYZZZZ
Version EF.eVerordnungsContainer	5	BCD	0x0020000000	Versionsnummer der Datei EF.eVerordnungsContainer im Format XXXYYYZZZZ
<ol style="list-style-type: none"> Das Informationselement Timestamp wird mit dem Zeitstempel des Produktionszeitpunktes vorbelegt Sollte beim Lesen dieser Datei festgestellt werden, dass sie komplett mit Nullen vorbelegt ist, ist anzunehmen, dass es sich um eine uninitialisierte Datei handelt. In diesem Fall soll sie mit den obigen Initialwerten vorbelegt werden. Ein Update weiterer Dateien ist damit nicht verbunden. 				

Es wird hier darauf hingewiesen, dass momentan keine Update-Prozedur für die Daten in der Datei EF.StatusVerordnungen definiert ist, da bisher nur diese eine gültige Version existiert. Sollte sich das Format der Container für eVerordnungen ändern, wird eine entsprechende Update-Prozedur veröffentlicht.

5 Notfalldaten

5.1 Einleitung

Die Notfalldaten werden auf der eGK in zwei Dateien abgelegt:

- Notfalldaten (EF.eNotfalldaten)
- NFD Status (EF.StatusNotfalldaten)

Die Datei EF.StatusNotfalldaten wird genutzt, um das Datum der letzten Aktualisierung, die Version und einen Status zu sichern. Der Status zeigt an, ob der letzte Aktualisierungsprozess vollständig durchgelaufen ist und die Daten technisch konsistent sind. Schreibabbrüche während der Aktualisierung lassen sich somit eindeutig erkennen.

5.2 Notfalldaten (EF.eNotfalldaten)

Im Container EF.eNotfalldaten werden die Daten zur Notfallversorgung entsprechend der Facharchitektur NFD (siehe [gemFA_NFDM]) in signierter Form als XML-Dokument abgespeichert.

Tabelle 10 - Struktur der Datei EF.eNotfalldaten

Informationselement	Länge in Byte	Typ	Initialwert	Bemerkung
Länge NFD	2	BINÄR	0	Siehe 1.
NFD	variabel	BINÄR	0	Siehe 2.

1. Gibt die Länge des Feldes NFD in Byte an. Liegen keine Notfalldaten NFD vor, so wird das Feld Länge mit dezimal 0 belegt um das Feld NFD als leer zu markieren. Der Maximalwert ergibt sich aus der in [gemSpec_eGK_P2] definierten Dateigröße für Notfalldaten abzüglich dem Speicherplatzbedarf für das Längengebiet (2 Byte)

2. Die Notfalldaten NFD selbst werden als XML-Daten gemäß vorgegebenem XML-Schema zusammen mit der Signatur des Arztes, gzip-komprimiert innerhalb der Datei abgelegt. Der zu verwendende Zeichensatz für die fachlichen Inhalte ist ISO8859-15.

5.3 Notfalldaten Status (EF.StatusNotfalldaten)

Die Datei EF.StatusNotfalldaten beinhaltet eine flache Datenstruktur mit 25 Byte Länge, kodiert in ISO8859-15.

Tabelle 11 - Struktur der Datei EF.StatusNotfalldaten

Informationselement	Länge in Byte	Typ	Initialwert	Bemerkung
Status	1	ALPHA	Siehe 1.	„1“ = Transaktionen offen „0“ = keine Transaktionen offen
Timestamp	14	ALPHA	Siehe 1.	Timestamp des letzten Schreibvorganges im Format YYYYMMDDhhmmss
Version	5	BCD	Siehe 1.	Versionsnummer der NFD Struktur im Format XXXYYYZZZZ, wird beim ersten Schreiben der Notfalldaten initial gesetzt
Reserviert	5	BINÄR	Siehe 1.	
1. Alle Informationselemente werden zum Personalisierungszeitpunkt mit NULL (0x00) vorbelegt.				

6 Protokolldaten

6.1 Protokolldaten (EF.Logging)

Die Datei EF.Logging wird mit 50 Records fester Satzlänge rotierend beschrieben. Die Datei EF.Logging ist auf der eGK als zyklische Datei nach dem FIFO-Prinzip umgesetzt, so dass hier keine Informationen zum Füllstand der Datei beschrieben oder verwaltet werden.

Tabelle 12 - Struktur der Datei EF.Logging

Informationselement	Länge in Byte	Typ	Initialwert	Bemerkung
Timestamp	6	BCD	0	Zeitpunkt des Datenzugriffs im Format YYYYMMDDhhmm
Component-ID	10	BCD	0	Optional (wenn vorhanden): EU-Resolution 190 konforme ICCSN, z. B. der SM-K/SM-AK des Konnektors, der den Protokolleintrag ausgelöst hat
Actor-ID	10	BCD	0	Optional (wenn vorhanden): EU-Resolution 190 konforme ICCSN, z.B. der HBA des zugreifenden Akteurs
Actor-Name	10	ALPHA	0x00	Name des zugreifenden Akteurs
Actor-Type	1	BINÄR	0	0 – Heilberufler 255 – Versicherter Der restliche Wertebereich ist zur weiteren Verwendung reserviert.
Data-Type	3	ALPHA	0x00	Objekttyp, z. B.: „VSD“ - Versichertenstammdaten „VOD“ - Verordnungen „NFD“ – Notfalldaten „VFA“ – Verwaltung freiw. Anwendungen
Data-Subtype	1	BINÄR	0	<u>NFD, VSD:</u> 0 - Kein Subtyp <u>VOD:</u> 1 – Arzneimittelverordnung <u>VFA:</u> 1 – Notfalldaten 2 – Arzneimitteltherapiesicherheit Diese Werte sind nicht normativ und werden in den entsprechenden Facharchitekturen definiert.
Data-Object-ID	2	BINÄR	0	Fachanwendungsspezifische Referenz auf das betroffene Datenobjekt. <u>VOD:</u> 1..8 – Record-Nummer im EF.eRezeptTicket-Container (siehe [gem-FA_VODM])

				Diese Werte sind nicht normativ und werden in den entsprechenden Facharchitekturen definiert.
Type of Access	1	ALPHA	0x00	<p>“R” – Read “W” - Write/Modify/Update „C“ – Create „D“ – Delete</p> <p>Diese Werte sind nicht normativ und werden in den entsprechenden Facharchitekturen definiert.</p>
Code of Action	2	BINÄR	0	Code des fachlichen Use-Cases, in dessen Rahmen der Zugriff erfolgt ist (siehe entsprechende Facharchitekturen VOD, VSD, NOT, VFA, ADV)

7 Dienstverweis

7.1 Dienstverweis (EF.Verweis)

Die Datei EF.Verweis beinhaltet 10 Records. Die Records werden genutzt, um den zur Einwilligungen des Versicherten zu eGK-basierten und zu telematikbasierten freiwilligen Anwendungen dazugehörigen Dienstverweise aufzunehmen. Ein Dienstverweis besteht hierbei aus dem Diensttyp und der Dienstinstanz in der Telematikinfrastruktur. Bei eGK-basierten Anwendungen ist die Dienstinstanz leer, da alle relevanten Daten auf der eGK selbst gespeichert werden und somit keine Referenz auf eine Komponente in der Telematikinfrastruktur abgelegt werden muss.

Durch die Record-Struktur dieser Datei lässt sich das fachlich geforderte Verbergen von Einwilligungen durch den „Aktivierung/ Deaktivierungs“-Mechanismus der eGK auf einzelne Records abbilden [gemFA ADV].

Eine feste Zuordnung von Diensttyp zu Record-Position existiert nicht, um die Anzahl der möglichen Diensttypen nicht unnötig auf 10 zu beschränken. Zu beachten ist, dass die Reihenfolge der Records, bezogen auf die betrachtete freiwillige Anwendung, dieselbe wie in EF.Einwilligung sein MUSS.

Tabelle 13 - Struktur der Datei EF.Verweis

Informationselement	Länge in Byte	Typ	Initialwert	Bemerkung
Diensttyp (1)	4	ALPHA	0x00	Siehe 1.
Dienstinstanz (1)	16	ALPHA	0x00	Siehe 2.
~~~~~				
Diensttyp (10)	4	ALPHA	0x00	Siehe 1.
Dienstinstanz (10)	16	ALPHA	0x00	Siehe 2.
<ol style="list-style-type: none"> <li>1. 0x00 := unbenutzter Record                      „AMTS“ := Arzneimitteltherapiesicherheit                      „NFD“ := Notfalldaten</li> <li>2. Die Dienstinstanz beinhaltet einen für den Versicherten lesbaren Namen, der vom SDS in eine URL umsetzbar ist. Für eGK-basierte Anwendungen wird die Dienstinstanz auf 0x00 gesetzt.</li> </ol>				

---

## 8 Einwilligung

---

### 8.1 Einleitung

Die Datei EF.Einwilligung beinhaltet 10 Records. Die Records werden genutzt, um die Einwilligungen des Versicherten zu eGK-basierten und zu telematikbasierten freiwilligen Anwendungen aufzunehmen.

In jedem Record werden der Dienstyp, der Status der Einwilligung, das Datum der Einwilligung und der Name des an der letzten Einwilligung zur konkreten freiwilligen Anwendung beteiligten Heilberufers abgelegt. Diese Daten dienen der Information des Versicherten.

In jedem Record wird zusätzlich Raum für einen Transaktionsstatus vorgehalten.

Durch die Record-Struktur dieser Datei lässt sich das fachlich geforderte Verbergen von Einwilligungen durch den „Aktivierung/ Deaktivierungs“-Mechanismus der eGK auf einzelne Records abbilden [gemFA ADV].

Eine feste Zuordnung von Dienstyp zu Record-Position existiert nicht, um die Anzahl der möglichen Dienstypen nicht unnötig auf 10 zu beschränken. Zu beachten ist, dass die Reihenfolge der Records, bezogen auf die betrachtete freiwillige Anwendung, dieselbe wie in EF.Verweis sein MUSS.

## 8.2 Einwilligung (EF.Einwilligung)

Tabelle 14 - Struktur der Datei EF.Einwilligung

Informationselement	Länge in Byte	Typ	Initialwert	Bemerkung
Diensttyp (1)	4	ALPHA	0x00	Siehe 3.
Name Heilberufler (1)	60	ALPHA	0x00	Name des Heilberuflers, der <b>den Status der Einwilligung</b> eingetragen hat. Der Name wird dem Attribut commonName des AUT Zertifikates des HBA entnommen. Siehe 1.
Timestamp (1)	4	BCD	0	Timestamp <b>der Eintragung</b> im Format YYYYMMDD
Flag (1)	1	Binär	0	Siehe 2.
Diensttyp (10)	4	ALPHA	0x00	Siehe 3.
Name Heilberufler (10)	60	ALPHA	0x00	Name des Heilberuflers, der <b>den Status der Einwilligung</b> eingetragen hat. Der Name wird dem Attribut commonName des AUT Zertifikates des HBA entnommen. Siehe 1.
Timestamp (10)	4	BCD	0	Timestamp <b>der Eintragung</b> im Format YYYYMMDD
Flag (10)	1	Binär	0	Siehe 2.
<p>1. 0x00 := unbenutzter Record</p> <p>2. Die Flagliste beinhaltet 8 Flags (b8..b1), wobei b8 das höchstwertige Bit ist. Bei Ausgabe der Karte sind alle Werte auf 0 zu setzen.</p> <pre> b8 b7 b6 b5 b4 b3 b2 b1       x  x  x      Transaktionsstatus:                         keine offene Transaktion = 0x0,                         weitere Transaktionsstati siehe [gemFA VFA]                         Einwilligung erteilt=1, nicht erteilt=0                         Reserviert       x  x  x  x </pre> <p>3. 0x00 := unbenutzter Record  „AMTS“ := Arzneimitteltherapiesicherheit  „NFD“ := Notfalldaten</p>				

Anmerkung: Die Zugriffsbedingungen auf die Datei EF.Einwilligung ist in der eGK-Spezifikation Teil 2 [gemSpec_eGK_P2] im Anhang B spezifiziert.

---

## 9 Version

---

### 9.1 Einleitung

Die Datei EF.Version beinhaltet 4 Records fester Länge. 3 Records dienen der Aufnahme der Version der eGK Spezifikationen, denen die Karte entspricht. Der letzte Record ist für weitere Anwendungen reserviert.

### 9.2 Version (EF.Version)

**Tabelle 15 - Struktur der Datei EF.Version**

Informationselement	Länge in Byte	Typ	Initialwert	Bemerkung
Version der unterstützten eGK Spec. Teil 1	5	BCD	0	Siehe 1.
Version der unterstützten eGK Spec. Teil 2	5	BCD	0	Siehe 1.
Version der unterstützten Speicherstrukturen [gemeGK_Fach]	5	BCD	0	Siehe 2.
Reserviert	5		0	

1. Versionsnummern der Dokumente [gemSpec_eGK_P1] und [gemSpec_eGK_P2] im Format XXXYYYZZZZ zu dem die Karte kompatibel ist

2. Versionsnummer dieses Dokumentes im Format XXXYYYZZZZ zu dem die Karte kompatibel ist, in diesem Fall 1.6.0

---

## 10 Testkennzeichen

---

### 10.1 Einleitung

Die Datei EF.TTN dient zur Aufnahme des Testkennzeichens. Das Testkennzeichen speichert die Teilnahme des Versicherten über die Speicherung von Testregion, des Testabschnitts und der Teststufe. Damit kann die Differenzierung der Testteilnehmer nach Releases geprüft, automatisch ausgewertet und dem Primärsystem angezeigt werden. Die Datei EF.TTN beinhaltet 5 Records fester Länge, d.h. es kann die gleichzeitige Teilnahme an maximal 5 verschiedenen Testphasen gespeichert werden.

### 10.2 Testkennzeichen (EF.TTN)

**Tabelle 16 - Struktur der Datei EF.TTN**

Informationselement	Länge in Byte	Typ	Initialwert	Bemerkung
Region	8	ALPHA	0	Testregion nach [RVO2006], §7. Kann eine der folgenden Zeichenketten beinhalten: BO-E, FL, HN, IN, LOEB-ZI, TR, WOB
Abschnitt	1	BINÄR	0	Funktionsumfang der Testung nach [RVO2006], §4 im Binärformat.
Stufe	1	BINÄR	0	Teststufe entsprechend [RVO2006], §5 im Binärformat.
Release	5	BCD	0	Releasebezeichnung im Format XXXYYYZZZZ

## Anhang A Netto-Speicherplatzbedarf der eGK

Datum		Speicherbedarf [Byte]
MF	ATR	32
	GDO	12
	DIR	190
	3DES (3 Paare)	144
	Version	20
	PIN (2x)	16
	PUK (2x)	16
	CVC	Zertifikat eGK
	Zertifikat CA	331
	Öffentliche Schlüssel der Root-CA	260
HCA	PD (Persönliche Versichertendaten)	850
	VD (Allgemeine Versicherungsdaten)	1.250
	GVD (Geschützte Versichertendaten)	450
	StatusVD	25
	Display Message	8
	Einwilligung	690
	Logging	2.300
	eVerordnungenTickets	1.320
	eVerordnungenContainer	30.000
	StatusVerordnungen	25
	Verweis	200
	eNotfalldaten	9.000
	StatusNotfalldaten	25
	<b>TTN</b>	<b>75</b>
QES	QES-Zertifikat inkl. öffentl.Schlüssel	1.900
	PIN	8
	PUK	8
eSign		0
	AUT-Zertifikat inkl. öffentl.Schlüssel	1.900
	AUTN-Zertifikat inkl. öffentl.Schlüssel	1.900
	ENC-Zertifikat inkl. öffentl.Schlüssel	1.900
	ENCV-Zertifikat inkl. öffentl.Schlüssel	1.900
	Display Message	8
CIAeSign	CIA-Info	23
	Summe	<b>57.127</b>

### Anmerkungen:

Betriebssystemspezifische Speicherplatzanforderungen und Verwaltungsoverheads werden in dieser Liste nicht aufgeführt. Dazu gehören die Speicherplatzanforderungen der EF.ARR und der privaten Schlüssel. Zusätzlich sind auch keine Zusatzanforderungen enthalten, die durch die Vorbereitung zum Nachladen der QES entstehen. Diese müssen (optionsspezifisch) hinzugerechnet werden.



Der Netto-Speicherbedarf überschreitet nicht die aktuell für die eGK Generation 1 vorgesehene Grenze von 58 KByte = 59392 Byte.

Der Container für Notfalldaten wurde für die Aufnahme der Erklärung zur Organspende angepasst.

Falls eine eGK keine qualifizierte Signatur bekommen soll, kann der dafür vorgesehene Bedarf von 1900 Byte von der Gesamtsumme abgezogen werden.

---

## Anhang B

---

### B1 Abkürzungen

Kürzel	Erläuterung
ADV	Anwendungen des Versicherten
ALPHA	Datentyp siehe Tabelle 2
BCD	Datentyp siehe Tabelle 2
BMG	Bundesministerium für Gesundheit
CDM	Common Data Modell
CM	Kartenmanagement
eGK	Elektronische Gesundheitskarte
FK	Fachkonzept
GKV	Gesetzliche Krankenversicherung
IK	Institutionskennzeichen
KVK	Krankenversichertenkarte
MKT	Multifunktionales Kartenterminal
NFDM	Notfalldatenmanagement
BINÄR	Datentyp siehe Tabelle 2
PKV	Private Krankenversicherung
SGB	Sozialgesetzbuch
UC	Use Case (Anwendungsfall)
UML	Unified Modelling Language
VOD	Verordnungsdaten
VODM	Verordnungsdatenmanagement
VSD	Versichertenstammdaten
VSDM	Versichertenstammdatenmanagement
XML	Extensible Markup Language

## B2 Glossar

Das zentrale Projektglossar wird als eigenständiges Dokument zur Verfügung gestellt.

## B3 Abbildungsverzeichnis

Abbildung 1 Speicherkonzept für eVerordnungen auf der eGK .....	14
Abbildung 2 Defragmentierung der Datei EF.eVerordnungsContainer .....	15
Abbildung 3 Verhinderung von Integritätsproblemen bei Defragmentierung .....	16
Abbildung 4 Zustandsdiagramm Zertifikatsreferenz im Konnektor .....	18

## B4 Tabellenverzeichnis

Tabelle 1: Fachkonzepte (FK) und Facharchitekturen (FA) zur Einführung der Gesundheitskarte .....	7
Tabelle 2 - Definition der Datentypen .....	8
Tabelle 3 - Struktur der Datei EF.VD .....	10
Tabelle 4 - Struktur der Datei EF.GVD .....	11
Tabelle 5 - Struktur der Datei EF.PD .....	11
Tabelle 6 - Struktur der Datei EF.StatusVD .....	12
Tabelle 7 - Struktur der Datei EF.eVerordnungTickets .....	20
Tabelle 8 - Struktur der Datei EF.eVerordnungsContainer .....	21
Tabelle 9 - Struktur der Datei EF.StatusVerordnungen .....	22
Tabelle 10 - Struktur der Datei EF.eNotfalldaten .....	23
Tabelle 11 - Struktur der Datei EF.StatusNotfalldaten .....	24
Tabelle 12 - Struktur der Datei EF.Logging .....	25
Tabelle 13 - Struktur der Datei EF.Verweis .....	27
Tabelle 14 - Struktur der Datei EF.Einwilligung .....	29
Tabelle 15 - Struktur der Datei EF.Version .....	30
Tabelle 16 - Struktur der Datei EF.TTN .....	31

## B5 Referenzierte Dokumente

[Quelle]	Herausgeber (Erscheinungsdatum): Titel
[gemFA_ADV]	gematik (06.02.2007): Einführung der Gesundheitskarte - Facharchitektur Anwendungen des Versicherten (ADV) Version 1.1.0
[gemFA_CMSeGK]	gematik (19.03.2008): Einführung der Gesundheitskarte - Facharchitektur Kartenmanagement eGK Version 1.5.0, <a href="http://www.gematik.de">www.gematik.de</a>
[gemFA_NFDM]	gematik (04.03.2008): Einführung der Gesundheitskarte - Facharchitektur Daten für die Notfallversorgung (Notfalldaten) Version 1.5.0, <a href="http://www.gematik.de">www.gematik.de</a>
[gemFA_VFA]	gematik (Draft 2007): Einführung der Gesundheitskarte - Facharchitektur Verwaltung freiwilliger Anwendungen (in Vorbereitung)
[gemFA_VODM]	gematik (20.03.2008): Einführung der Gesundheitskarte - Facharchitektur Verordnungsdatenmanagement Version 1.4.0
[gemFA_VSDM]	gematik (28.02.2008): Einführung der Gesundheitskarte - Facharchitektur Versichertenstammdatenmanagement Version 2.7.0
[gemFK_CMSeGK]	gematik (19.03.2008): Einführung der Gesundheitskarte - Fachkonzept Kartenmanagement der eGK Version 1.2.0
[gemFK_NFDM]	gematik (02.10.2007): Einführung der Gesundheitskarte - Fachkonzept Daten für die Notfallversorgung Version 1.3.0, <a href="http://www.gematik.de">www.gematik.de</a>
[gemFK_VODM]	gematik (20.03.2008): Einführung der Gesundheitskarte - Fachkonzept Verordnungsdatenmanagement, Version 2.5.0, <a href="http://www.gematik.de">www.gematik.de</a>
[gemFK_VSDM]	gematik (28.02.2008): Einführung der Gesundheitskarte - Fachkonzept Versichertenstammdatenmanagement, Version 2.7.0, <a href="http://www.gematik.de">www.gematik.de</a>
[gemSpec_eGK_P1]	gematik (20.03.2008): Einführung der Gesundheitskarte – Die Spezifikation elektronische Gesundheitskarte; Teil 1 – Spezifikation der elektrischen Schnittstelle Version 2.2.0, <a href="http://www.gematik.de">www.gematik.de</a>
[gemSpec_eGK_P2]	gematik (25.03.2008): Einführung der Gesundheitskarte – Die Spezifikation elektronische Gesundheitskarte ; Teil 2 – Grundlegende Applikationen Version 2.2.0, <a href="http://www.gematik.de">www.gematik.de</a>
[gemSpec_SG]	gematik (Draft 2008): Einführung der Gesundheitskarte – Spezifikation Generierung und Aufteilung des privaten Schlüssels PrK.CH.ENC des Versicherten (in Vorbereitung)
[ISO8859-15]	ISO8859-15:1999 Information technology -- 8-bit single-byte coded graphic character sets - - Part 15: Latin alphabet No. 9
[ISO8601]	ISO 8601:2000 Data elements and interchange formats – Information interchange – Representation of dates and times

[Quelle]	Herausgeber (Erscheinungsdatum): Titel
[RFC2119]	RFC 2119 (März 1997): Key words for use in RFCs to Indicate Requirement Levels, <a href="http://www.ietf.org/rfc/rfc2119.txt">http://www.ietf.org/rfc/rfc2119.txt</a>
[RVO2006]	Bundesgesetzblatt I (2006) vom 10.10.2006, Seite 2199 ff.: Verordnung über Testmaßnahmen für die Einführung der elektronischen Gesund- heitskarte in der Fassung der Bekanntmachung vom 5. Oktober 2006