

Elektronische Gesundheitskarte und Telematikinfrastruktur

Spezifikation Fachlogik der Fachanwendung NFDM in KTR-AdV (FLA-NFDM)

Version: 1.1.0
Revision: 17623
Stand: 14.05.2018
Status: freigegeben
Klassifizierung: öffentlich
Referenzierung: gemSpec_FLN_NFDM

Dokumentinformationen

Änderungen zur Vorversion

Einarbeitung von Änderungen lt. P15.2

Dokumentenhistorie

Version	Stand	Kap./ Seite	Grund der Änderung, besondere Hinweise	Bearbeitung
1.0.0	02.08.17		freigegeben	gematik
			Einarbeitung lt. Änderungsliste	gematik
1.1.0	14.05.18		freigegeben	gematik

Inhaltsverzeichnis

1	Einordnung des Dokumentes	5
1.1	Zielsetzung	5
1.2	Zielgruppe	5
1.3	Geltungsbereich	5
1.4	Abgrenzungen	5
1.5	Methodik.....	6
2	Systemüberblick	8
3	Systemkontext	9
3.1	Akteure und Rollen.....	9
3.2	Nachbarsysteme	9
4	Zerlegung des Produkttyps	10
5	Übergreifende Festlegungen	11
5.1	Fehlerbehandlung.....	11
5.2	Berechtigungen	11
5.2.1	Fachliche Rollen	11
5.2.2	Berechtigung	11
5.3	Unterstützte Generationen der eGK	12
5.4	Versionierung	12
5.5	Protokollierung	12
5.5.1	Zugriffsprotokolleinträge auf der eGK	12
5.6	Datenspeicherung	13
5.7	Meldungen am Kartenterminal.....	13
5.8	Übergreifende Vorbedingungen	14
5.9	Übergreifende Nachbedingungen.....	15
6	Funktionsmerkmale.....	16
6.1	Schnittstelle I_DPE_Management	16
6.1.1	Schnittstellendefinition	16
6.1.1.1	ReadDPE	16
6.1.1.2	WriteDPE.....	19
6.1.1.3	EraseDPE.....	23
6.1.2	Umsetzung	26
6.1.2.1	ReadDPE	26

6.1.2.2	WriteDPE.....	29
6.1.2.3	EraseDPE.....	34
6.2	Schnittstelle I_COPY_DATA	38
6.2.1	Schnittstellendefinition	38
6.2.1.1	GetData	38
6.2.1.2	PutData	41
6.2.2	Umsetzung	45
6.2.2.1	GetData	45
6.2.2.2	PutData	45
6.3	Artefakte	47
6.3.1	DPE-Speicherstruktur der eGK.....	47
6.3.2	Der DPE auf der eGK	47
6.4	Testunterstützung	47
6.5	Hardwaremerkmale.....	47
7	Informationsmodell	48
8	Verteilungssicht.....	49
9	Anhang A – Verzeichnisse	50
9.1	– Abkürzungen.....	50
9.2	– Glossar	51
9.3	– Abbildungsverzeichnis.....	51
9.4	– Tabellenverzeichnis.....	51
9.5	– Referenzierte Dokumente.....	52
9.5.1	– Dokumente der gematik.....	52
9.5.2	– Weitere Dokumente	52
10	Anhang B.....	54
10.1	– Fehlermeldungen.....	54

1 Einordnung des Dokumentes

1.1 Zielsetzung

Die vorliegende Spezifikation definiert die spezifischen Anforderungen für die Fachlogik der Fachanwendung NFDM in KTR-AdV (FLA-NFDM).

FLA-NFDM ist diejenige Komponente der Fachanwendung NFDM, welche die Anwendungsfälle der Fachanwendung NFDM und die Anwendungsfälle des Leistungsmerkmals „Datenübertragung bei Kartentausch“ der Fachanwendung AdV in KTR-AdV auf Seite der AdV-App umsetzt. Der Funktionsumfang wird definiert durch das Leistungsmerkmal NFDM-L_2 („Persönliche Erklärungen NFDM“) der Fachanwendung NFDM. Dieses wurde in [gemSysL_NFDM] definiert und stellt KTR-AdV eine Grundfunktionalität zur Verwaltung von Datensätzen für persönliche Erklärungen (DPE) zur Verfügung.

1.2 Zielgruppe

Das Dokument richtet sich an Hersteller der KTR-AdV, die die Fachlogik der Fachanwendung NFDM umsetzen.

1.3 Geltungsbereich

Dieses Dokument enthält normative Festlegungen zur Telematikinfrastruktur des deutschen Gesundheitswesens. Der Gültigkeitszeitraum der vorliegenden Version und deren Anwendung in Zulassungs- oder Abnahmeverfahren wird durch die gematik GmbH in gesonderten Dokumenten (z. B. Dokumentenlandkarte, Produkttypsteckbrief, Leistungsbeschreibung) festgelegt und bekannt gegeben.

Schutzrechts-/Patentrechtshinweis

Die nachfolgende Spezifikation ist von der gematik allein unter technischen Gesichtspunkten erstellt worden. Im Einzelfall kann nicht ausgeschlossen werden, dass die Implementierung der Spezifikation in technische Schutzrechte Dritter eingreift. Es ist allein Sache des Anbieters oder Herstellers, durch geeignete Maßnahmen dafür Sorge zu tragen, dass von ihm aufgrund der Spezifikation angebotene Produkte und/oder Leistungen nicht gegen Schutzrechte Dritter verstoßen und sich ggf. die erforderlichen Erlaubnisse/Lizenzen von den betroffenen Schutzrechtsinhabern einzuholen. Die gematik GmbH übernimmt insofern keinerlei Gewährleistungen.

1.4 Abgrenzungen

Spezifiziert werden in dem Dokument die von FLA-NFDM bereitgestellten logischen Schnittstellen. Es werden Plattformbausteine aus KTR-AdV gemäß [gemSpec_KTR-AdV] verwendet, die Zugriff auf produktübergreifende Plattformleistungen definieren.

Die vollständige Anforderungslage für FLA-NFDM ergibt sich aus weiteren Konzept- und Spezifikationsdokumenten. Diese sind in dem Produkttypsteckbrief der KTR-AdV verzeichnet, da FLA-NFDM integraler Bestandteil der KTR-AdV ist.

1.5 Methodik

Anforderungen als Ausdruck normativer Festlegungen werden durch eine eindeutige ID in eckigen Klammern sowie die dem RFC 2119 [RFC2119] entsprechenden, in Großbuchstaben geschriebenen deutschen Schlüsselworte MUSS, DARF NICHT, SOLL, SOLL NICHT, KANN gekennzeichnet.

Sie werden im Dokument wie folgt dargestellt:

<AFO-ID> - <Titel der Afo>

Text / Beschreibung

[<=]

Dabei umfasst die Anforderung sämtliche innerhalb der Textmarken angeführten Inhalte.

Hinweise zur Nomenklatur:

- Schnittstellen-, Operations-, Parameter- und Dateinamen sowie Bezeichner für Objekte auf der elektronischen Gesundheitskarte (eGK), Namen der referenzierten Technischen Use Cases (TUCs) des Konnektors und Extensible-Markup-Language(XML)-Elemente oder -Attribute werden in diesem Dokument in nicht-proportionaler Schriftart gesetzt.
- Hexadezimale Zahlen und Oktett-Strings werden in Hochkommata eingeschlossen (z. B. '2F 03').

Bezeichner für Objekte der eGK:

Der Bezeichner der in [gemSpec_eGK_ObjSys] definierten Objekte der eGK dient dazu, das Objekt innerhalb der Spezifikation zu identifizieren. Adressieren lassen sich die Objekte auf der eGK über deren Attribut `pwdIdentifier` (für Passwortobjekte), `fileIdentifier` (für Elementary Files (EFs)) oder `applicationIdentifier` (für Dedicated Files (DFs)). Ein File Identifier (FID) /Application Identifier (AID)/Password Identifier (PID) muss nur innerhalb eines Ordners eindeutig sein. Aus Gründen der Lesbarkeit und um Unterschiede zwischen eGK G2 und eGK G2.1 darzustellen, werden in dieser Spezifikation Bezeichner eingeführt, für die in der folgenden Tabelle sowohl der Bezeichner als auch der FID/AID/PID mit dem übergeordneten Ordner (überg. DF) angegeben sind.

Tabelle 1: Tab_FLA_NFDM_001 Verwendete Bezeichner für Objekte der eGK

Objekt		G 2.0	G 2.1
Bezeichner	FID o. AID o. PID	Bezeichner des überg. DF	Bezeichner des überg. DF
MF	'D276 0001 4480 00'		
DF.ESIGN	'A000000167	MF	MF

	455349474E'		
EF.Version2	'2F 10'	MF	MF
EF.C.CH.AUTN.R2048	'C5 00'	DF.ESIGN	DF.ESIGN
DF.HCA	'D276000001 02'	MF	MF
EF.Logging	'D0 06'	DF.HCA	DF.HCA
DF.NFD	'D276 0001 4407'	DF.HCA	DF.HCA
EF.NFD	'D0 10'	DF.NFD	DF.NFD
EF.StatusNFD	'D0 0E'	DF.NFD	DF.NFD
DF.DPE	'D276 0001 4408'	DF.HCA	DF.HCA
EF.DPE	'D0 1B'	DF.DPE	DF.DPE
EF.StatusDPE	'D0 18'	DF.DPE	DF.DPE
PIN.CH	'01'	MF	MF
MRPIN.NFD	'03'	DF.NFD	MF
MRPIN.NFD_READ	'07'	DF.NFD	MF
MRPIN.DPE	'04'	DF.DPE	MF
MRPIN.DPE_READ	'08'	DF.DPE	nicht vorhanden

2 Systemüberblick

FLA-NFDM ist Bestandteil des Produkttyps KTR-AdV der Fachanwendung AdV. Die Systemzerlegung der Fachanwendung AdV ist in [gemSysL_AdV#2.1] definiert.

Der Produkttyp KTR-AdV wird gebildet aus einem AdV-Server mit zugehöriger AdV-App. Die AdV-App beinhaltet Plattformbausteine, Serverproxy und Fachlogik, zu welcher FLA-NFDM gehört. Die Zerlegung des Produkttyps KTR-AdV wird in [gemSpec_KTR-AdV#4] definiert.

FLA-NFDM nutzt die dezentrale TI-Plattform in KTR-AdV durch Aufruf der Plattformbausteine gemäß [gemSpec_KTR-AdV#Anhang B]. FLA-NFDM ist integraler Bestandteil der KTR-AdV Umgebung (App und AdV-Server), d.h. die Spezifikationen KTR-AdV (als Plattformkomponente) und FLA-NFDM sind zwar getrennt, werden aber von einem Hersteller als eine Funktionalität aus einem AdV-Server mit zugehöriger AdV-App umgesetzt. Die inneren (rein logischen) Schnittstellen zwischen FLA-NFDM und KTR-AdV sind von außen nicht erkennbar.

FLA-NFDM stellt seine Anwendungsfälle über folgende Schnittstellen der KTR-AdV zur Verfügung:

- I_DPE_Management für das Leistungsmerkmal NFDM-L_2 der Fachanwendung NFDM
- I_COPY_DATA für das Leistungsmerkmal „Datenübertragung bei Kartentausch“ der Fachanwendung AdV

Die Operationen der Schnittstelle I_DPE_Management bilden die Anwendungsfälle des Leistungsmerkmals NFDM-L_2 gemäß folgender Tabelle ab.

Tabelle 2: Tab_FLA_NFDM_019 Abbildung NFDM-Anwendungsfälle auf Operationen des FLA-NFDM

Anwendungsfall aus [gemSysL_NFDM]	Operation FLA-NFDM
I_DPE_Management	
NFDM_UC_200 DPE auf eGK schreiben	WriteDPE
NFDM_UC_201 DPE von eGK lesen	ReadDPE
NFDM_UC_202 DPE von eGK löschen	EraseDPE

3 Systemkontext

3.1 Akteure und Rollen

FLA-NFDM ist Bestandteil des Produkttyps KTR-AdV und unterliegt damit der Definition für Akteure und Rollen von KTR-AdV gemäß [gemSpec_KTR-AdV#3.1].

Der Nutzer, als natürliche Person, welche die Anwendungsfälle NFDM und AdV in der KTR-AdV startet, agiert in der Rolle „Versicherter“.

3.2 Nachbarsysteme

FLA-NFDM ist integraler Bestandteil von KTR-AdV. Auf der logischen Ebene in KTR-AdV ist FLA-NFDM Teil der Fachlogik der AdV-App. Die Nachbarsysteme der KTR-AdV sind in [gemSpec_KTR-AdV#3.2] dargestellt.

4 Zerlegung des Produkttyps

Eine weitere Untergliederung der Aufbaustruktur von FLA-NFDM ist nicht erforderlich.

5 Übergreifende Festlegungen

5.1 Fehlerbehandlung

Tritt bei der Ausführung einer Operation von FLA-NFDM ein Fehler auf, der zum Abbruch der Operation führt, so wird dieser an die aufrufende AdV-App der KTR-AdV gemeldet.

Für das Fehlermanagement gelten neben den hier aufgeführten spezifischen Anforderungen für das FLA-NFDM die Anforderungen aus Kapitel 3 der übergreifenden Spezifikation [gemSpec_OM].

NFDM-A_2390 - Fehlermeldungen inkl. Trace

FLA-NFDM MUSS sicherstellen, dass Fehlermeldungen einen Trace beinhalten, der den auftretenden Fehler in der Software von der Fehlerursache zurück bis zum auslösenden Operationsaufruf vollständig nachvollziehbar macht.

[<=]

NFDM-A_2405 - Trace ohne Implementierungsdetails

FLA-NFDM DARF NICHT im Trace der Fehlermeldungen Informationen übermitteln, die Rückschlüsse über die Art und Weise der Implementierung zulassen (z.B. Teile des Programm-Stack-Traces).

[<=]

NFDM-A_2391 - Vorgaben spezifische Fehlermeldungen

FLA-NFDM MUSS sicherstellen, dass alle spezifischen Fehlermeldungen den Vorgaben gemäß Tabelle „Tab_FLA_NFDM_018 – Fehlermeldungen“ genügen.

[<=]

5.2 Berechtigungen

5.2.1 Fachliche Rollen

Der fachliche Akteur (hier Versicherter) ruft fachliche Anwendungsfälle von NFDM zur Anzeige und Bearbeitung der auf seiner eGK gespeicherten Daten auf. Der Versicherte erhält mittels der fachlichen Rolle, die technisch durch das Zugriffsprofil seiner eGK repräsentiert wird, die Autorisierung zur Ausführung des jeweiligen Anwendungsfalls.

5.2.2 Berechtigung

FLA-NFDM spezifiziert die Fachlogik NFDM. Die Berechtigungen zum Zugriff auf die eGK, die für die Anwendungsfälle NFDM erforderlich sind werden durch KTR-AdV gesteuert und sind nicht Bestandteil von FLA-NFDM.

5.3 Unterstützte Generationen der eGK

NFDM-A_2392 - Unterstützte Generationen der eGK

FLA-NFDM MUSS alle Versionen der eGK der Generationen G2 und höher unterstützen, die ihm bekannt sind. Das FLA-NFDM DARF NICHT die Generationen G0, G1 und G1+ unterstützen.

Falls die ermittelte Kartenversion kleiner als alle zu unterstützenden Kartenversionen ist oder die ermittelte Kartenversion nicht bekannt und kleiner als die größte zu unterstützende Kartenversion ist, MUSS FLA-NFDM von einer inkompatiblen Karte ausgehen und die weitere Verarbeitung der Karte unmittelbar abbrechen.

Falls die ermittelte Kartenversion größer als alle FLA-NFDM bekannten Kartenversionen ist, MUSS FLA-NFDM von einer kompatiblen Karte ausgehen und versuchen, diese zu verarbeiten.

[<=]

Die zu unterstützenden Versionen der Karten ergeben sich aus den in der zu dieser Spezifikation gehörigen Dokumentenlandkarte aufgeführten zugelassenen Produkttypversionen.

5.4 Versionierung

FLA-NFDM ist integraler Bestandteil des Produkttyps KTR-AdV und unterliegt der Versionierung von KTR-AdV. Deshalb ist für FLA-NFDM keine separate Versionierung vorgesehen.

5.5 Protokollierung

Für die Protokollierung von FLA-NFDM in KTR-AdV gelten die Anforderungen zum Logging aus [gemSpec_KTR-AdV#5.2]. Darüberhinaus gelten in FLA-NFDM die folgenden Anforderungen.

NFDM-A_2393 - Verbot Protokollierung medizinischer Daten

FLA-NFDM DARF NICHT medizinische Daten des Versicherten protokollieren.

[<=]

NFDM-A_2394 - Verbot Protokollierung Schlüsselmaterial

FLA-NFDM DARF NICHT geheimes Schlüsselmaterial protokollieren.

[<=]

NFDM-A_2395 - Verbot Protokollierung personenbezogener Daten

FLA-NFDM DARF NICHT personenbezogene Daten des Versicherten protokollieren.

[<=]

5.5.1 Zugriffsprotokolleinträge auf der eGK

NFDM-A_2396 - Zugriffsprotokollierung auf der eGK

FLA-NFDM MUSS die in Tabelle „Tab_FLA_NFDM_002 – Werte der Zugriffsprotokolleinträge auf der eGK“ definierten Werte für die Informationselemente eines Zugriffsprotokolleintrags auf der eGK verwenden.

Tabelle 3: Tab_FLA_NFDM_002 – Werte der Zugriffsprotokolleinträge auf der eGK

Operation	Data Type	Type of Access		Timestamp, Actor-ID, Actor-Name
ReadDPE	c	R	Lesen	gemäß Tabelle „Tab_Karten_Fach_TIP_010_Struktur“ in [gemSpec_Karten_Fach_TIP#4.1]
WriteDPE	c	W	Schreiben	
EraseDPE	c	E	Löschen	
GetData	c	R	Lesen des Datensatzes zum Kopieren	
PutData	c	C	Kopierter Datensatz der alten eGK	

Hinweis:
Data Type „c“ bedeutet, dass der Kartenzugriff auf Daten der Anwendung DPE in der KTR-AdV-Umgebung erfolgt.

[<=]

5.6 Datenspeicherung

NFDM-A_2397 - Verbot der persistenten Speicherung medizinischer Daten

FLA-NFDM DARF NICHT medizinische Daten des Versicherten persistent speichern.

[<=]

5.7 Meldungen am Kartenterminal

NFDM-A_2399 - Terminalanzeigen für PIN-Eingabe

FLA-NFDM SOLL im Rahmen der PIN-Eingaben bei Aufruf der Umgebungsoperation ENV_TUC_SECRET_INPUT am Kartenterminal bei Operationen der Schnittstelle I_DPE_Management und I_COPY_DATA die in Tabelle „Tabelle 4: Tab_FLA_NFDM_003 – Terminal-Anzeigen für PIN-Eingabe“ definierten Werte als Parameter für die PIN-Verifizierung verwenden.

Tab_FLA_NFDM_003 – Terminal-Anzeigen für PIN-Eingabe“ definierten Werte als Parameter für die PIN-Verifizierung verwenden.

[<=]

Wenn es sich um ein Kartenterminal ohne Anzeige handelt, dann kann die Anforderung NFDM-A_2399 nicht umgesetzt werden. Die Anzeige wird dann ausschließlich durch die AdV-App in KTR-AdV gemäß „Tabelle 4: Tab_FLA_NFDM_003 – Terminal-Anzeigen für PIN-Eingabe“ realisiert.

Tabelle 4: Tab_FLA_NFDM_003 – Terminal-Anzeigen für PIN-Eingabe

Operation	Terminalanzeige
-----------	-----------------

ReadDPE WriteDPE EraseDPE GetDPE PutDPE	PIN•0x0Bfür•0x0Bpers. •Erklärungen 0x0FVersicherten-PIN:
<p>0x0B mögliches Trennzeichen: Ein Display eines Kartenterminals hat physikalisch bedingt eine begrenzte Anzahl von Zeichen je Zeile. "0x0B" bezeichnet die Position im anzuzeigenden Text, an welcher der Zeilenumbruch möglich ist, für den Fall dass das Display nicht den gesamten Text in einer Zeile darstellen kann. Ein Zeilenumbruch an anderen Stellen im Text ist nicht erlaubt.</p> <p>0x0F Trennzeichen zwischen Nachricht und PIN-Prompt) Leerzeichen werden als „•“ dargestellt.</p> <p>Die Zeilenumbrüche in der Spalte „Terminalanzeige“ sind editorisch bedingt.</p>	

5.8 Übergreifende Vorbedingungen

Vorbedingungen sind Bedingungen, die erfüllt sein müssen, bevor eine Operation von FLA-NFDM aufgerufen werden kann. Die Bedingungen müssen von der aufrufenden Komponente in KTR-AdV durchgesetzt werden.

Die folgenden „Übergreifenden Vorbedingungen“ gelten für für alle Operationen der in Kapitel 6 definierten Services und werden von der aufrufenden Komponente vor dem Aufruf der Operationen der Schnittstelle I_DPE_Management und I_COPY_DATA in KTR-AdV geprüft:

Tabelle 5: Tab_FLA_NFDM_004 – Übergreifende Vorbedingungen für FLA-NFDM

ID	Bedingung
ÜV1	Die übergebenen Werte der Parameter sind gültig gemäß der Signatur der jeweiligen Operation.
ÜV2	Die Gesundheitsanwendung der eGK (DF.HCA) ist nicht gesperrt (= ist nicht deaktiviert).
ÜV3	Die beteiligte eGK hat keine ältere Versionsnummer als die der Generation 2.
ÜV4	Die beteiligten Smartcards sind echt.
ÜV5	Der DPE auf der eGK ist sichtbar, d. h. der Ordner DF.DPE der eGK muss als Wert des Attributs lifeCycleStatus den Wert „active“ haben

Darüberhinaus können in den Operationen weitere Vorbedingungen definiert werden.

5.9 Übergreifende Nachbedingungen

Erfolgsbedingungen sind Bedingungen, die erfüllt sein müssen, damit eine Operation von FLA-NFDM erfolgreich durchlaufen werden kann. Jede der Bedingungen wird von den Operationen im Rahmen der Verarbeitung überprüft.

Folgende „Übergreifende Erfolgsbedingungen“ gelten für alle Operationen der in Kapitel 6 definierten Services. Sie werden in allen Operationsspezifikationen des Kapitel 6 referenziert.

Tabelle 6: Tab_FLA_NFDM_005 – Übergreifende Erfolgsbedingungen

ID	Bedingung
-	-

Folgende „Übergreifende Nachbedingungen“ gelten für alle Operationen der in Kapitel 6 definierten Services. Sie werden in allen Operationsspezifikationen des Kapitel 6 referenziert.

Tabelle 7: Tab_FLA_NFDM_006 – Übergreifende Nachbedingungen

ID	Bedingung
ÜN1	Ist die ermittelte Version eGK cardVersion G2.1 oder höher dann wurde ein Zugriffsprotokolleintrag gemäß Tabelle „Tab_FLA_NFDM_002 – Werte der Zugriffsprotokolleinträge auf der eGK“ als Record im <code>EF.Logging</code> auf der eGK geschrieben, andernfalls wurde kein Protokolleintrag geschrieben (sondern erfolgt in KTR-AdV).

6 Funktionsmerkmale

In diesem Kapitel werden die Schnittstellen von FLA-NFDM definiert. Für jede Operation wird das an der jeweiligen Schnittstelle sichtbare und damit testbare Verhalten normativ in tabellarischer Form spezifiziert.

Die Unterkapitel „Umsetzung“ beschreiben die Umsetzung der Operationsabläufe der jeweiligen Schnittstelle in der KTR-AdV Umgebung mittels der aufzurufenden Plattformbausteine, die KTR-AdV zur Verfügung stellt, oder internen Operationen von FLA-NFDM. Tabellarisch wird jeder Aktion der Aktivitätsdiagramme entweder ein bzw. mehrere Plattformbausteine des KTR-AdV zugeordnet oder – falls keine aktionsrelevante Dienstfunktionalität bereitgestellt wird – eine interne Funktion benannt und deren Aufgabe beschrieben. Werden nicht explizit im Fehlerfalle zurückzugebende Fehlermeldungen genannt, werden die Fehlermeldungen der aufgerufenen Plattformbausteine zurückgegeben. Der in KTR-AdV realisierte Card Proxy gemäß [gemSpec_KTR-AdV] hat das Wissen über die fachspezifischen Anteile der jeweiligen Smartcard und setzt die erforderliche Funktionalität um. Deshalb muss FLA-NFDM in KTR-AdV folgende Aspekte nicht spezifizieren:

- wo die Artefakte auf einer Karte liegen,
- welche Zugriffsrechte zum Ausführen von Aktionen erforderlich (bspw. Verifikation einer PIN, externe Authentisierung eines CV-Zertifikates mittels Card-2-Card) sind,
- wie erforderliche Zugriffsrechte erworben werden können, und
- wie auf Daten zugegriffen oder gewisse Aktionen ausgeführt werden

6.1 Schnittstelle I_DPE_Management

FLA-NFDM stellt mit der Schnittstelle `I_DPE_Management` Operationen in KTR-AdV zur Verfügung, die die fachlichen Anwendungsfälle von NFDM zum Lesen, Schreiben und Löschen des Datensatzes „Persönliche Erklärungen“ auf der eGK unterstützen. Es handelt sich hierbei um eine interne Schnittstelle, deren Signatur und Verhalten definiert wird, jedoch keine Festlegungen hinsichtlich Implementierung getroffen werden.

Die Schnittstelle `I_DPE_Management` umfasst die 3 Operationen:

- `ReadDPE`: DPE von eGK lesen
- `WriteDPE`: DPE auf eGK schreiben
- `EraseDPE`: DPE von eGK löschen

6.1.1 Schnittstellendefinition

6.1.1.1 ReadDPE

NFDM-A_2400 - ReadDPE

FLA-NFDM MUSS für KTR-AdV die Operation `ReadDPE` gemäß Tabelle 8:

`Tab_FLA_NFDM_007` – Operation `ReadDPE` anbieten.

Tabelle 8: `Tab_FLA_NFDM_007` – Operation `ReadDPE`

Name	ReadDPE	
Beschreibung	Die Operation liest den DPE im Informationselement <code>DPE</code> der Datei <code>EF.DPE</code> von der eGK und gibt ihn über den Parameter <code>DPEDocument</code> an den Aufrufer zurück.	
Vorbedingungen	In KTR-AdV werden vor Aufruf dieser Operation die übergreifenden Vorbedingungen aus Tabelle „Tab_FLA_NFDM_004 – Übergreifende Vorbedingungen für FLA-NFDM“ geprüft.	
Erfolgsbedingungen	Die Operation MUSS die folgenden Erfolgsbedingungen überprüfen.	
	ID	Bedingung
	E1	Der auf der eGK gespeicherte DPE ist technisch konsistent, d. h. der Wert des Informationselement <code>Status</code> der Datei <code>EF.StatusDPE</code> der eGK ist „0“.
	E2	Die Version der internen Speicherstruktur (s. 6.3.1) der Dateien der Anwendung „Datensatz Persönliche Erklärungen“ der eGK (DPE-Speicherstruktur) wird von FLA-NFDM unterstützt.
	E3	Es ist ein gemäß [RFC1952] gzip-komprimierter DPE auf der eGK im Informationselement <code>DPE</code> der Datei <code>EF.DPE</code> gemäß [gemSpec_eGK_Fach_NFDM#3.1] gespeichert, d. h. das Informationselement <code>Länge DPE</code> der Datei <code>EF.DPE</code> der eGK hat einen Wert ungleich '00 00'.
	E4	Der auf der eGK gespeicherte DPE ist valide gegen das XML-Schema für den DPE (s. [gemSpec_InfoNFDM#5]).
Aufrufparameter	keine	
Rückgabe	<code>DPEDocument</code>	Von der eGK des Versicherten gelesener, dekomprimierter und validierter DPE gemäß gemSpec_KTR-AdV
Nachbedingungen	Falls alle Erfolgsbedingungen erfüllt sind, MUSS die Operation die übergreifenden Nachbedingungen gemäß Tabelle „Tab_FLA_NFDM_006 – Übergreifende Nachbedingungen“ und die folgenden erfüllen.	
	ID	Bedingung
	N1	Der DPE (Ausgabeparameter <code>DPEDocument</code>) ist dekomprimiert.

Ablauf	<p>Der Ablauf der Operation <code>ReadDPE</code>, der das definierte Außenverhalten abbildet, ist im Aktivitätsdiagramm der Abbildung 1: Abb_FLA_NFDM_001 – Ablauf <code>ReadDPE</code> modelliert. Die Umsetzung des Ablaufs mittels Plattformbausteinen von KTR-AdV und interner Operationen von FLA-NFDM spezifiziert Tabelle 11: Tab_FLA_NFDM_010 – Umsetzung Ablaufaktivitäten <code>ReadDPE</code>. Der Hersteller kann von der Umsetzung bzw. den spezifizierten Abläufen (z. B. zum Zwecke der Performanceoptimierung) abweichen, falls dadurch das definierte Außenverhalten der Operation gewährleistet bleibt.</p>																				
Fehlermeldungen	<p>Für die generischen Fehlermeldungen finden sich die Attribute <code>ErrorType</code>, <code>Severity</code>, <code>Fehlertext</code>, <code>Befüllung Details</code> und <code>Auslösende Bedingung</code> in [gemSpec_OM#3.2.2]. Für die spezifischen Fehlermeldungen sind die Attribute <code>ErrorType</code>, <code>Severity</code>, <code>Fehlertext</code> und <code>Befüllung Details</code> in Tabelle „Tab_FLA_NFDM_018 – Fehlermeldungen“ definiert.</p> <p>Generische Fehlermeldungen</p> <table> <tr> <th>Code</th><th>Befüllung Details</th></tr> <tr> <td>108</td><td>Der Detailtext KANN den Fehler näher beschreiben.</td></tr> <tr> <td>111</td><td>Der Detailtext KANN den Fehler näher beschreiben.</td></tr> </table> <p>Spezifische Fehlermeldungen</p> <table> <tr> <th>Code</th><th>Auslösende Bedingung</th></tr> <tr> <td>5103</td><td>E1 ist nicht erfüllt.</td></tr> <tr> <td>5104</td><td>E2 ist nicht erfüllt.</td></tr> <tr> <td>5106</td><td>Die Dekomprimierung des DPE ist gescheitert.</td></tr> <tr> <td>5114</td><td>E4 ist nicht erfüllt.</td></tr> <tr> <td>5121</td><td>E3 ist nicht erfüllt.</td></tr> <tr> <td>5500</td><td>Jegliches fehlerhafte Verhalten, das nicht durch die anderen Fehlermeldungen erfasst wird.</td></tr> </table>	Code	Befüllung Details	108	Der Detailtext KANN den Fehler näher beschreiben.	111	Der Detailtext KANN den Fehler näher beschreiben.	Code	Auslösende Bedingung	5103	E1 ist nicht erfüllt.	5104	E2 ist nicht erfüllt.	5106	Die Dekomprimierung des DPE ist gescheitert.	5114	E4 ist nicht erfüllt.	5121	E3 ist nicht erfüllt.	5500	Jegliches fehlerhafte Verhalten, das nicht durch die anderen Fehlermeldungen erfasst wird.
Code	Befüllung Details																				
108	Der Detailtext KANN den Fehler näher beschreiben.																				
111	Der Detailtext KANN den Fehler näher beschreiben.																				
Code	Auslösende Bedingung																				
5103	E1 ist nicht erfüllt.																				
5104	E2 ist nicht erfüllt.																				
5106	Die Dekomprimierung des DPE ist gescheitert.																				
5114	E4 ist nicht erfüllt.																				
5121	E3 ist nicht erfüllt.																				
5500	Jegliches fehlerhafte Verhalten, das nicht durch die anderen Fehlermeldungen erfasst wird.																				

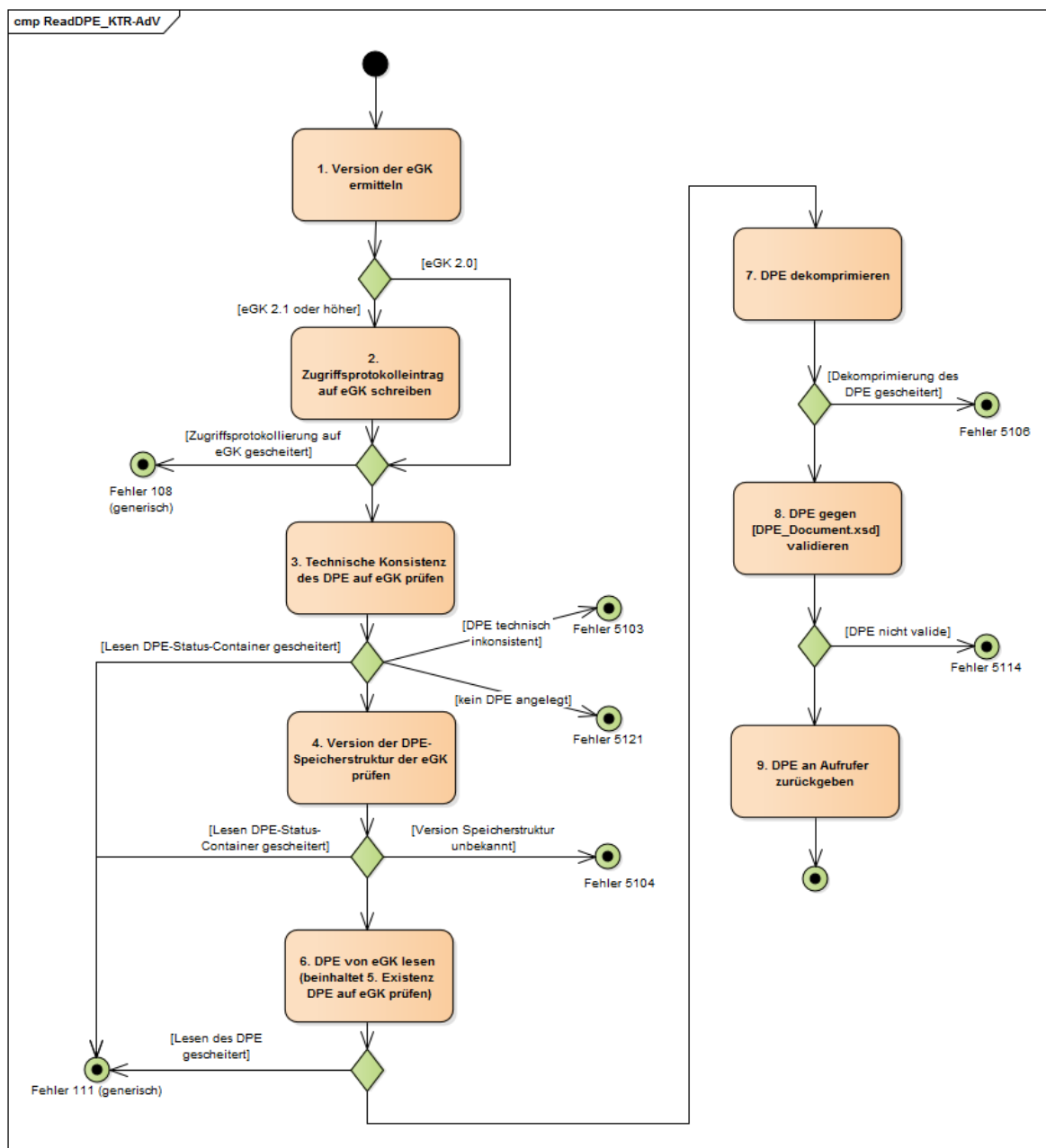


Abbildung 1: Abb_FLA_NFDM_001 – Ablauf ReadDPE

[<=]

6.1.1.2 WriteDPE

NFDM-A_2401 - WriteDPE

FLA-NFDM MUSS für KTR-AdV die Operation `WriteDPE` gemäß Tabelle 9: Tab_FLA_NFDM_008 – Operation `WriteDPE` anbieten.

Tabelle 9: Tab_FLA_NFDM_008 – Operation `WriteDPE`

Name	WriteDPE
------	----------

Beschreibung	Die Operation schreibt den ihr im Parameter <code>DPEDocument</code> übergebenen DPE auf die eGK in das Informationselement <code>DPE</code> der Datei <code>EF.DPE</code>	
Vorbedingungen	In KTR-AdV werden vor Aufruf dieser Operation die übergreifenden Vorbedingungen aus Tabelle „Tab_FLA_NFDM_004 – Übergreifende Vorbedingungen für FLA-NFDM“ und folgende Vorbedingung geprüft.	
	ID	Bedingung
	V1	Das Zertifikat <code>C.CH.AUTN</code> der eGK ist zeitlich gültig und nicht gesperrt.
Erfolgsbedingungen	Die Operation MUSS die folgenden Erfolgsbedingungen überprüfen.	
	ID	Bedingung
	E1	Der übergebene DPE ist valide gegen das XML-Schema für den DPE (s. <code>[gemSpec_InfoNFDM#5]</code>).
	E2	Der Versicherte der eGK ist mit dem Versicherten des DPE identisch, d. h. die Versicherten-ID (die ersten zehn unveränderbaren Stellen der Krankenversicherungsnummer (KVNR)) im Feld <code>organizationalUnitName</code> des Subject Distinguished Name (SubjectDN) des Authentisierungszertifikats der eGK (gespeichert in der Datei <code>EF.C.CH.AUTN.R2048</code>) ist die gleiche wie die im DPE im Element <code>Versicherten_ID</code> gespeicherte. <i>Anmerkung: Es gibt zwei <code>organizationalUnitName</code>-Felder im SubjectDN. Das zehnstellige, alphanumerische Feld beinhaltet die Versicherten-ID (unveränderbarer Teil der Krankenversicherungsnummer), das andere, neunstellige numerische Feld das Institutionskennzeichen der Krankenversicherungsnummer (s. <code>[gemSpec_PKI#5.1.3.1]</code>).</i>
	E3	Der DPE ist nicht größer als der auf der eGK im Informationselement <code>DPE</code> der Datei <code>EF.DPE</code> der eGK zur Verfügung stehende Speicherplatz.
	Aufrufparameter	<code>DPEDocument</code> Auf die eGK des Versicherten zu schreibender DPE gemäß <code>gemSpec_KTR-AdV</code>
Rückgabe	keine	
Nachbedingungen	Falls alle Erfolgsbedingungen erfüllt sind, MUSS die Operation die übergreifenden Nachbedingungen gemäß Tabelle „Tab_FLA_NFDM_006 – Übergreifende Nachbedingungen“ und die folgenden erfüllen.	
	ID	Bedingung
	N1	Die Größe des DPE in Oktett ist im Informationselement <code>Länge DPE</code> der Datei <code>EF.DPE</code> der eGK gemäß

		[gemSpec_eGK_Fach_NFDM#3.1] gespeichert.												
	N2	Der übergebene DPE ist gemäß [RFC1952] gzip-komprimiert auf der eGK im Informationselement DPE der Datei EF.DPE gemäß [gemSpec_eGK_Fach_NFDM#3.1] gespeichert.												
	N3	Der Wert des Informationselements Timestamp der Datei EF.StatusDPE der eGK ist aktualisiert.												
	N4	Der Wert des Informationselements Version_Speicherstruktur der Datei EF.StatusDPE der eGK ist aktualisiert mit einer gemäß [gemSpec_eGK_Fach_NFDM#3.2] gültigen Versionsnummer.												
	N5	Der Wert des Informationselement Status der Datei EF.StatusDPE der eGK ist „0“.												
Ablauf	Der Ablauf der Operation WriteDPE, der das definierte Außenverhalten abbildet, ist im Aktivitätsdiagramm der Abbildung „Abb_FLA_NFDM_002 – Ablauf WriteDPE“ modelliert. Die Umsetzung des Ablaufs mittels Plattformbausteinen von KTR-AdV und interner Operationen FLA-NFDM spezifiziert Tabelle 12: Tab_FLA_NFDM_011 – Umsetzung Ablaufaktivitäten WriteDPE. Der Hersteller kann von der Umsetzung bzw. den spezifizierten Abläufen (z. B. zum Zwecke der Performanceoptimierung) abweichen, falls dadurch das definierte Außenverhalten der Operation gewährleistet bleibt.													
Fehlermeldungen	<p>Für die generischen Fehlermeldungen finden sich die Attribute ErrorType, Severity, Fehlertext, Befüllung Details und Auslösende Bedingung in [gemSpec_OM#3.2.2]. Für die spezifischen Fehlermeldungen sind die Attribute ErrorType, Severity, Fehlertext und Befüllung Details in Tabelle „Tab_FLA_NFDM_018 – Fehlermeldungen“ definiert.</p> <p>Generische Fehlermeldungen</p> <table><tr><th>Code</th><th>Befüllung Details</th></tr><tr><td>108</td><td>Der Detailtext KANN den Fehler näher beschreiben.</td></tr><tr><td>112</td><td>Der Detailtext KANN den Fehler näher beschreiben.</td></tr></table> <p>Spezifische Fehlermeldungen</p> <table><tr><th>Code</th><th>Auslösende Bedingung</th></tr><tr><td>5000</td><td>Die eGK ist defekt.</td></tr><tr><td>5108</td><td>E2 ist nicht erfüllt.</td></tr></table>		Code	Befüllung Details	108	Der Detailtext KANN den Fehler näher beschreiben.	112	Der Detailtext KANN den Fehler näher beschreiben.	Code	Auslösende Bedingung	5000	Die eGK ist defekt.	5108	E2 ist nicht erfüllt.
Code	Befüllung Details													
108	Der Detailtext KANN den Fehler näher beschreiben.													
112	Der Detailtext KANN den Fehler näher beschreiben.													
Code	Auslösende Bedingung													
5000	Die eGK ist defekt.													
5108	E2 ist nicht erfüllt.													

	5110	Die Komprimierung des DPE ist gescheitert.
	5113	E3 ist nicht erfüllt
	5114	E1 nicht erfüllt.
	5500	Jegliches fehlerhafte Verhalten, das nicht durch die anderen Fehlermeldungen erfasst wird.

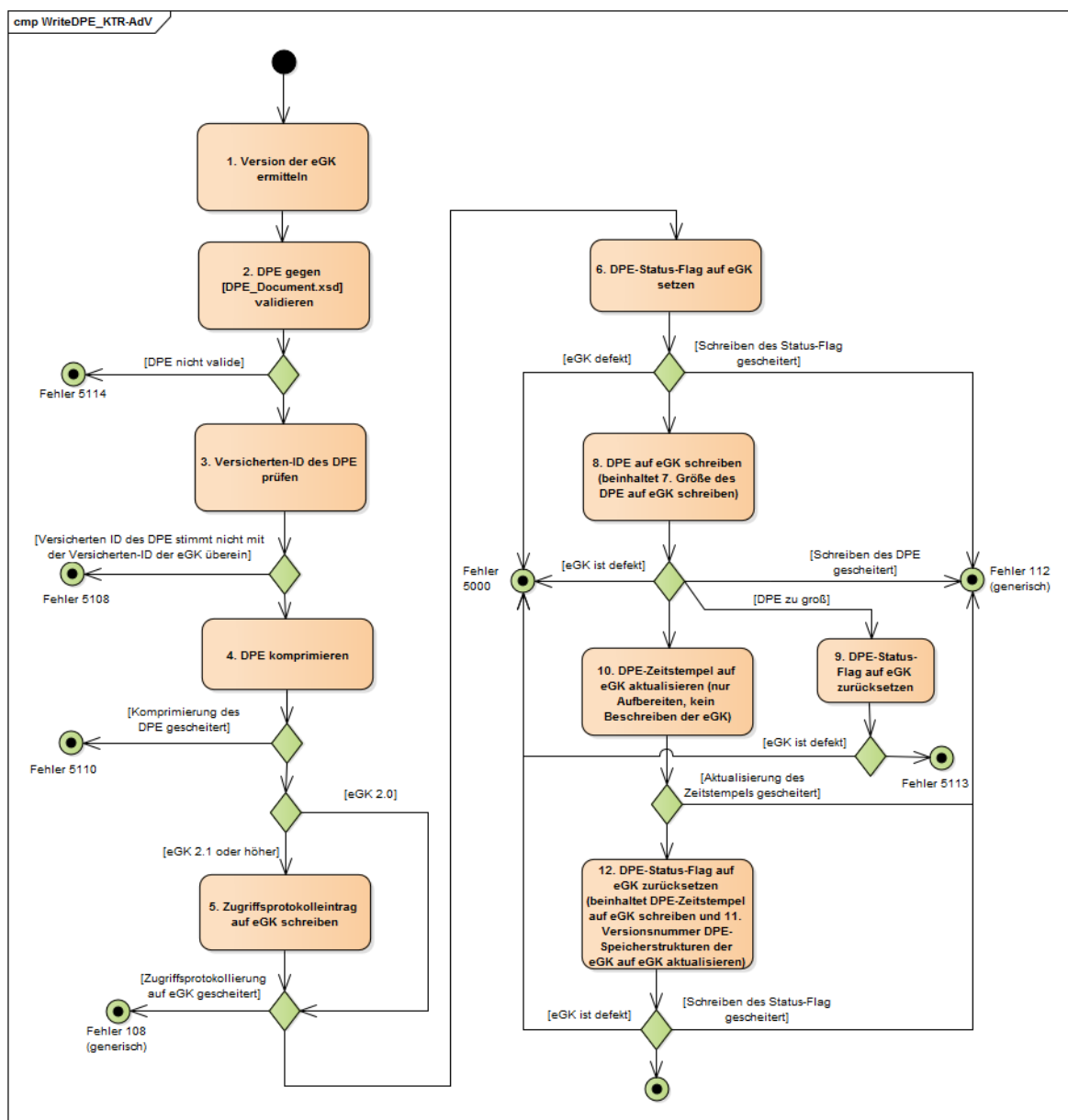


Abbildung 2: Abb_FLA_NFDM_002 – Ablauf WriteDPE

[<=]

6.1.1.3 EraseDPE

NFDM-A_2402 - EraseDPE

FLA-NFDM MUSS für KTR-AdV die Operation EraseDPE gemäß Tabelle 10:

Tab_FLA_NFDM_009 – Operation EraseDPE anbieten.

Tabelle 10: Tab_FLA_NFDM_009 – Operation EraseDPE

Name	EraseDPE	
Beschreibung	Die Operation löscht den DPE im Informationselement DPE der Datei EF.DPE der eGK.	
Vorbedingungen	In KTR-AdV werden vor Aufruf dieser Operation die übergreifenden Vorbedingungen aus Tabelle „Tab_FLA_NFDM_004 – Übergreifende Vorbedingungen für FLA-NFDM“ geprüft.	
Erfolgsbedingungen	Die Operation MUSS die folgenden Erfolgsbedingungen überprüfen.	
	ID	Bedingung
	E1	Die Version der internen Speicherstruktur (s. 6.3.1) der Dateien der Anwendung „Datensatz Persönliche Erklärungen“ der eGK (DPE-Speicherstruktur) wird FLA-NFDM unterstützt.
Aufrufparameter	keine	
Rückgabe	keine	
Nachbedingungen	Falls alle Erfolgsbedingungen erfüllt sind, MUSS die Operation die übergreifenden Nachbedingungen gemäß Tabelle „Tab_FLA_NFDM_006 – Übergreifende Nachbedingungen“ und die folgenden erfüllen.	
	ID	Bedingung
	N1	Der Wert des Informationselements Länge DPE der Datei EF.DPE der eGK ist '00 00'.
	N2	Der DPE im Informationselement DPE der Datei EF.DPE der eGK ist gelöscht, d.h. alle ursprünglich vom DPE belegten Oktette sind mit '00' (NULL) überschrieben worden.
	N3	Der Wert des Informationselements Timestamp der Datei EF.StatusDPE der eGK ist aktualisiert.
	N4	Der Wert des Informationselements Version_Speicherstruktur der Datei EF.StatusDPE der eGK ist aktualisiert mit einer gemäß [gemSpec_eGK_Fach_NFDM#3.2] gültigen Versionsnummer.
	N5	Der Wert des Informationselements Status der Datei EF.StatusDPE der eGK ist „0“.

Ablauf	Der Ablauf der Operation EraseDPE, der das definierte Außenverhalten abbildet, ist im Aktivitätsdiagramm der Abbildung 3: Abb_FLA_NFDM_003 – Ablauf EraseDPE modelliert. Die Umsetzung des Ablaufs mittels Plattformbausteinen von KTR-AdV und interner Operationen von FLA-NFDM spezifiziert Tabelle 13: Tab_FLA_NFDM_012 – Umsetzung Ablaufaktivitäten EraseDPE. Der Hersteller kann von der Umsetzung bzw. den spezifizierten Abläufen (z. B. zum Zwecke der Performanceoptimierung) abweichen, falls dadurch das definierte Außenverhalten der Operation gewährleistet bleibt.	
Fehlermeldungen	Für die generischen Fehlermeldungen finden sich die Attribute ErrorType, Severity, Fehlertext, Befüllung Details und Auslösende Bedingung in [gemSpec_OM#3.2.2]. Für die spezifischen Fehlermeldungen sind die Attribute ErrorType, Severity, Fehlertext und Befüllung Details in Tabelle „Tab_FLA_NFDM_018 – Fehlermeldungen “ definiert.	
	Generische Fehlermeldungen	
	Code	Befüllung Details
	108	Der Detailtext KANN den Fehler näher beschreiben.
	112	Der Detailtext KANN den Fehler näher beschreiben.
	Spezifische Fehlermeldungen	
	Code	Auslösende Bedingung
	5000	Die eGK ist defekt.
	5104	E1 ist nicht erfüllt.
	5112	Löschen des DPE (technisch) gescheitert.
5500	Jegliches fehlerhafte Verhalten, das nicht durch die anderen Fehlermeldungen erfasst wird.	

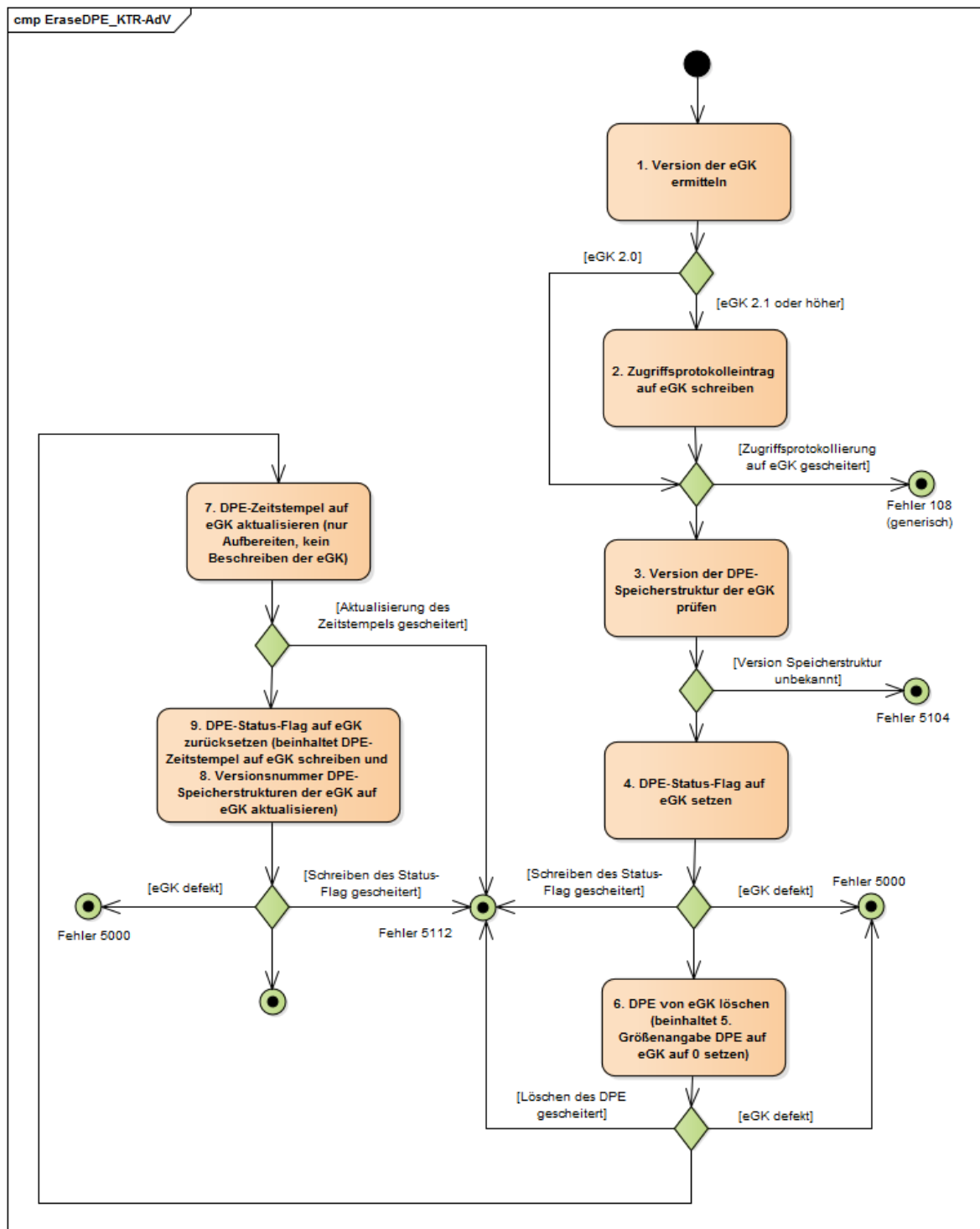


Abbildung 3: Abb_FLA_NFDM_003 – Ablauf EraseDPE

[<=]

6.1.2 Umsetzung

Die folgenden Unterkapitel beschreiben die Umsetzung der Operationsabläufe von `I_DPE_Management` in der KTR-AdV Umgebung.

6.1.2.1 ReadDPE

Tabelle 11: Tab_FLA_NFDM_010 – Umsetzung Ablaufaktivitäten ReadDPE

1	Version der eGK ermitteln	
	Information lt. Plattformbaustein <code>PL_TUC_CARD_INFORMATION</code>	
	Beschreibung	
	Für die eGK wird die Produkttypversion des COS ermittelt.	
2	Zugriffsprotokolleintrag auf eGK schreiben	
	Plattformbaustein <code>PL_TUC_EGK_APPEND_PROTOCOL</code>	
	Eingangsdaten	
	<code>cardVersion</code>	In 1 für eGK ermittelte Version
	<code>dataType</code>	gemäß Tabelle „Tab_FLA_NFDM_002 – Werte der Zugriffsprotokolleinträge auf der eGK“
	<code>accessType</code>	gemäß Tabelle „Tab_FLA_NFDM_002 – Werte der Zugriffsprotokolleinträge auf der eGK“
	Beschreibung	
	Ist die ermittelte Version eGK <code>cardVersion</code> G2.1 oder höher dann erfolgt das Schreiben des Zugriffsprotokolleintrags, andernfalls wird kein Protokolleintrag geschrieben (erfolgt durch KTR-AdV). Ein Zugriffsprotokolleintrag wird als Record in die Datei <code>EF.Logging</code> auf der eGK geschrieben. Tritt bei der Zugriffsprotokollierung ein Fehler auf, bricht die Operation mit Fehler 108 ab.	
3	Technische Konsistenz des DPE auf eGK prüfen	
3.1	Aufruf Plattformbaustein <code>PL_TUC_CARD_READ_FILE</code>	
	Eingangsdaten	
	<code>IDENTIFIKATOR</code>	<code>EF.StatusDPE</code>
	Beschreibung	

	Es werden alle Informationselemente der Datei EF.StatusDPE der eGK gelesen. Tritt dabei ein Lesefehler auf, bricht die Operation mit Fehler 111 ab.
3.2	NFDM:checkConsistency
	Eingangsdaten
	In 3.1 aus der Datei EF.StatusDPE gelesene Daten (= alle Informationselemente)
	Beschreibung
	Es wird überprüft, ob der Wert des Informationselements Status der Eingangsdaten „0“ ist. Ist der Status „0“, ist der DPE technisch konsistent und die Verarbeitung läuft weiter. Ist der Status „1“, ist der Fehler 5103 mit dem Wert des Informationselements Timestamp der Eingangsdaten im Detailtext zurückzugeben. Ist der Wert weder „0“ noch „1“ gemäß [gemSpec_eGK_Fach_NFDM#2.2], so wurde noch kein DPE angelegt. Die Operation bricht daher mit dem Fehler 5121 ab.
4	Version der DPE-Speicherstruktur der eGK prüfen
	NFDM:checkContainerVersion
	Eingangsdaten
	In 3.1 aus dem DPE-Statuscontainer der eGK gelesene Daten (= alle Informationselemente)
	Beschreibung
	Es wird überprüft, ob FLA-NFDM die im Informationselement Version_Speicherstruktur der Eingangsdaten gespeicherte Versionsnummer der DPE-Speicherstruktur der eGK bekannt ist. Ist dies nicht der Fall, bricht die Operation mit Fehler 5104 ab.
5	Existenz DPE auf eGK prüfen
	Die Prüfung, ob ein DPE auf der eGK existiert, erfolgt implizit in 6. In 6 werden alle Daten aus der Datei EF.DPE der eGK gelesen. Das Informationselement Länge_DPE enthält die Größe (Anzahl Oktette) des im Informationselement DPE gespeicherten DPE. Nur wenn ein Wert für dieses Informationselement gesetzt ist und dieser ungleich '00 00' ist, existiert ein DPE auf der eGK.
6	DPE von eGK lesen
6.1	Aufruf Plattformbaustein PL_TUC_CARD_READ_FILE

	Eingangsdaten		
	<table><tr><td>IDENTIFIKATOR</td><td>EF.DPE</td></tr></table>	IDENTIFIKATOR	EF.DPE
IDENTIFIKATOR	EF.DPE		
	Beschreibung		
	Es werden die Daten aus beiden Informationselementen (Länge DPE und DPE) der Datei EF.DPE der eGK gelesen. Tritt dabei ein Fehler auf, bricht die Operation mit Fehler 111 ab.		
6.2	NFDM:checkSize		
	Eingangsdaten		
	In 6.1 aus der Datei EF.DPE der eGK gelesene Daten (die beiden Informationselemente Länge DPE und DPE)		
	Beschreibung		
	Es wird das Informationselement Länge DPE der Eingangsdaten ausgewertet. Das Informationselement Länge DPE enthält die Größe (Anzahl Oktette) des im Informationselement DPE gespeicherten DPE. Ist die Größe '00 00', bedeutet dies, es existiert kein DPE auf der eGK. In diesem Fall bricht die Operation mit Fehler 5121 ab.		
6.3	NFDM:extractDpe		
	Eingangsdaten		
	In 5.1 aus der Datei EF.DPE der eGK gelesene Daten (die beiden Informationselemente Länge DPE und DPE)		
	Beschreibung		
	Die dem Informationselement Länge DPE folgende und der in 6.2 ausgewerteten Größe entsprechende Anzahl Oktette wird als DPE zur Weiterverarbeitung extrahiert. Tritt dabei ein Fehler auf, bricht die Operation mit Fehler 111 ab.		
7	DPE dekomprimieren		
	NFDM:decompress		
	Eingangsdaten		
	Der in 6.3 extrahierte DPE		
	Beschreibung		

	Der DPE wird dekomprimiert. Tritt dabei ein Fehler auf, bricht die Operation mit Fehler 5106 ab.
8	DPE gegen [DPE_Document.xsd] validieren
	NFDM:validateDocument
	Beschreibung
	Der DPE wird gegen das XML-Schema [DPE_Document.xsd] validiert. Ist der DPE nicht valide gegen das Schema, so bricht die Operation mit Fehler 5114 ab.
9	DPE an Aufrufer zurückgeben
	RückgabeDPEDocument

6.1.2.2 WriteDPE

Tabelle 12: Tab_FLA_NFDM_011 – Umsetzung Ablaufaktivitäten WriteDPE

1	Version der eGK ermitteln
	Information lt. Plattformbaustein PL_TUC_CARD_INFORMATION
	Beschreibung
	Für die eGK wird die Betriebssystemversion ermittelt und das Zertifikat C.CH.AUTN gelesen.
2	DPEDocument gegen [DPE_Document.xsd] validieren
	NFDM: validateDocument
	Beschreibung
	Der DPE wird gegen das XML-Schema [DPE_Document.xsd] validiert. Ist der DPE nicht valide gegen das Schema, so bricht die Operation mit Fehler 5114 ab.
3	Versicherten-ID des DPE prüfen
3.1	NFDM:extractDpeInsurantId
	Eingangsdaten
	DPEDocument

	Beschreibung	
	Die Versicherten-ID (Wert des Elements /DPEDocument/Persönliche_Erklärungen/DPE_Versicherter/Versicherter/Versicherten_ID) wird aus DPEDocument extrahiert.	
3.2	NFDM:extractSubjectDN	
	Beschreibung	
	Aus dem in 1 ermittelten Zertifikat C.CH.AUTN wird die SubjectDN extrahiert.	
3.3	NFDM:extractCertInsurantId	
	Beschreibung	
	<p>Aus der in 3.2 ermittelten SubjectDN wird die Versicherten-ID im Feld mit dem Namen organizationalUnitName extrahiert. Die Struktur des SubjectDN ist in [gemSpec_PKI#5.1.2] und [gemSpec_PKI#5.1.3.1] definiert.</p> <p>Anmerkung: Es gibt zwei organizationalUnitName-Felder. Das zehnstellige, alphanumerische Feld beinhaltet die Versicherten-ID (unveränderbarer Teil der Krankenversicherthenummer), das andere, neunstellige numerische Feld, das Institutionskennzeichen der Krankenversicherthenummer (s. [gemSpec_PKI#5.1.3.1]).</p>	
3.4	NFDM:checkInsurantIdsForEquality	
	Beschreibung	
	<p>Die in 3.3 aus dem DPE extrahierte Versicherten ID und die in 3.2 aus SubjectDN von C.CH.AUTN extrahierte Versicherten-ID werden auf Gleichheit getestet. Sind die beiden IDs nicht gleich, bricht die Operation mit Fehler 5108 ab.</p>	
4	DPE komprimieren	
	NFDM:compress	
	Beschreibung	
	<p>Das in 2 validierte DPEDocument wird komprimiert.</p> <p>Tritt in diesem Schritt ein Fehler auf, bricht die Operation mit Fehler 5110 ab.</p>	
5	Zugriffsprotokolleintrag auf eGK schreiben	
	Plattformbaustein PL_TUC_EGK_APPEND_PROTOCOL	
	Eingangsdaten	
	cardVersion	In 1 für eGK ermittelte Version

	dataType	gemäß Tabelle „Tab_FLA_NFDM_002 – Werte der Zugriffsprotokolleinträge auf der eGK“
	accessType	gemäß Tabelle „Tab_FLA_NFDM_002 – Werte der Zugriffsprotokolleinträge auf der eGK“
	Beschreibung	
	<p>Ist die ermittelte Version eGK <code>cardVersion</code> G2.1 oder höher dann erfolgt das Schreiben des Zugriffsprotokolleintrags, andernfalls wird kein Protokolleintrag geschrieben (erfolgt durch KTR-AdV).</p> <p>Ein Zugriffsprotokolleintrag wird als Record in die Datei <code>EF.Logging</code> auf der eGK geschrieben. Tritt bei der Zugriffsprotokollierung ein Fehler auf, bricht die Operation mit Fehler 108 ab.</p>	
6	DPE-Status-Flag auf eGK setzen	
	Plattformbaustein <code>PL_TUC_CARD_UPDATE_FILE</code>	
	Eingangsdaten	
	IDENTIFIKATOR	<code>EF.StatusDPE</code>
	OFFSET	0
	NEWDATA	1
	Beschreibung	
	<p>Zur „Transaktionssicherung“ wird der Wert des Informationselements <code>Status</code> der Datei <code>EF.StatusDPE</code> der eGK vor Beginn des Schreibvorgangs des DPE auf „1“ gesetzt.</p> <p>Liefert <code>PL_TUC_CARD_UPDATE_FILE</code> als Ergebnis <code>`MemoryFailure`</code>, bricht die Operation mit Fehler 5000 ab.</p> <p>In allen anderen Fehlerfällen bricht die Operation mit Fehler 112 ab.</p>	
7	Größe des DPE auf eGK schreiben	
7.1	Größe des DPE ermitteln	
	<code>NFDM:determineSize</code>	
	Eingangsdaten	
	Der in 4 komprimierte DPE	
	Beschreibung	

	Die Größe (Anzahl Oktette) des DPE wird bestimmt.	
7.2	Das Schreiben der in 7.1 ermittelten Größe des DPE auf die eGK erfolgt zusammen mit dem Schreiben des DPE in 8.	
8	DPE auf eGK schreiben	
8.1	NFDM:concatenate	
	Eingangsdaten	
	DPE und die in 7.1 ermittelte Größe (Anzahl Oktette) des DPE	
	Beschreibung	
	Die in 7.1 ermittelte Größenangabe wird konkateniert mit dem DPE. Die Struktur entspricht [gemSpec_eGK_Fach_NFDM#3.1]. Tritt hierbei ein Fehler auf, bricht die Operation mit Fehler 112 ab.	
8.2	Plattformbaustein PL_TUC_CARD_UPDATE_FILE	
	Eingangsdaten	
	IDENTIFIKATOR	EF.DPE
	Beschreibung	
	Die in 8.1 konkatenierte Oktettkette (Größenangabe und DPE) wird auf die eGK in die Datei EF.DPE geschrieben. Liefert PL_TUC_CARD_UPDATE_FILE als Ergebnis `MemoryFailure`, bricht die Operation mit Fehler 5000 ab. Liefert PL_TUC_CARD_UPDATE_FILE als Ergebnis `DataTooBig` (Größe des zu schreibenden DPE ist größer als der in der Datei EF.DPE entsprechend des Objektsystems der eGK zur Verfügung stehende Speicherplatz) wird Schritt 9 ausgeführt und die Operation mit dem Fehler 5113 abgebrochen. In allen anderen Fehlerfällen bricht die Operation mit Fehler 112 ab.	
9	DPE-Status-Flag auf eGK zurücksetzen	
	Falls die Größenprüfung beim Aufruf von PL_TUC_CARD_UPDATE_FILE in 8.2 mit Fehler `DataTooBig` endete.	
9.1	Plattformbaustein PL_TUC_CARD_UPDATE_FILE	
	Eingangsdaten	
	IDENTIFIKATOR	EF.StatusDPE
	OFFSET	0

	NEWDATA	0
	Beschreibung	
	<p>Mit Beendigung des Schreibvorgangs des DPE auf die eGK wird das DPE-Status-Flag (Wert des Informationselements <code>Status</code> der Datei <code>EF.StatusDPE</code> der eGK) wieder zurück auf „0“ gesetzt.</p> <p>Liefert <code>PL_TUC_CARD_UPDATE_FILE</code> als Ergebnis <code>`MemoryFailure`</code>, bricht die Operation mit Fehler 5000 ab.</p> <p>In allen anderen Fehlerfällen bricht die Operation mit Fehler 112 ab.</p>	
9.2	Die Operation bricht mit dem Fehler 5113 ab.	
	10 DPE-Zeitstempel auf eGK aktualisieren	
10.1	NFDM:getTime	
	Beschreibung	
	Liefere Systemzeit, die durch den Plattformbaustein <code>PL_TUC_NET_SYNC_TIME</code> bereitgestellt wird.	
10.2	NFDM:formatSystemTime	
	Eingangsdaten	
	Im vorherigen Teilschritt ermittelte Systemzeit	
	Beschreibung	
	<p>Die Systemzeit wird für das Schreiben in das Informationselement <code>Timestamp</code> der Datei <code>EF.StatusDPE</code> der eGK gemäß [gemSpec_eGK_Fach_NFDM#3.2] formatiert (14 Oktette; YYYYMMDDhhmmss).</p> <p>Das eigentliche Schreiben des DPE-Zeitstempels in das Informationselement <code>Timestamp</code> der Datei <code>EF.StatusDPE</code> der eGK erfolgt aus Performancegründen in 12.2. Dort wird die Oktettkette zur Befüllung der Datei <code>EF.StatusDPE</code> als Ganzes in die Datei <code>EF.StatusDPE</code> der eGK geschrieben.</p> <p>Tritt hierbei ein Fehler auf, bricht die Operation mit Fehler 112 ab.</p>	
11	Versionsnummer DPE-Speicherstruktur der eGK auf eGK aktualisieren	
	<p>Das Aktualisieren der Versionsnummer erfolgt implizit im nächsten Schritt. Dort wird die Oktettkette zur Befüllung der Datei <code>EF.StatusDPE</code> als Ganzes in die Datei <code>EF.StatusDPE</code> auf der eGK geschrieben.</p>	
	12 DPE-Status-Flag auf eGK zurücksetzen	
12.	NFDM:concatenate	

1	Eingangsdaten	
	In 10.2 formatierter DPE-Zeitstempel, Versionsnummer der DPE-Speicherstruktur der eGK, gemäß derer der DPE auf die eGK geschrieben wird.	
	Beschreibung Die Oktettkette für den DPE-Status-Container wird gemäß [gemSpec_eGK_Fach_NFDM#3.2] konkateniert, wobei das Informationselement <code>Version_XML</code> mit 5 Oktetten des Wertes '00' zu befüllen ist und das Informationselement <code>Status</code> den Wert „0“ erhält. Tritt hierbei ein Fehler auf, bricht die Operation mit Fehler 112 ab.	
12.	Plattformbaustein PL_TUC_CARD_UPDATE_FILE	
2	Eingangsdaten	
	IDENTIFIKATOR	EF.StatusDPE
	OFFSET	0
	NEWDATA	0
	Beschreibung Mit Beendigung des Schreibvorgangs des DPE auf die eGK wird das DPE-Status-Flag (Wert des Informationselements <code>Status</code> der Datei <code>EF.StatusDPE</code> der eGK) wieder zurück auf „0“ gesetzt und der DPE-Zeitstempel sowie die Versionsnummer der DPE-Speicherstruktur der eGK gemäß [gemSpec_eGK_Fach_NFDM#3.2] aktualisiert. Liefert PL_TUC_CARD_UPDATE_FILE als Ergebnis `MemoryFailure`, bricht die Operation mit Fehler 5000 ab. In allen anderen Fehlerfällen bricht die Operation mit Fehler 112 ab.	

6.1.2.3 EraseDPE

Tabelle 13: Tab_FLA_NFDM_012 – Umsetzung Ablaufaktivitäten EraseDPE

1	Version der eGK ermitteln
	Information lt. Plattformbaustein PL_TUC_CARD_INFORMATION
	Beschreibung
	Für die eGK wird die Betriebssystemversion ermittelt.
2	Zugriffsprotokolleintrag auf eGK schreiben

	Plattformbaustein PL_TUC_EGK_APPEND_PROTOCOL								
	Eingangsdaten <table> <tr> <td>cardVersion</td><td>In 1 für eGK ermittelte Version</td></tr> <tr> <td>dataType</td><td>gemäß Tabelle „Tab_FLA_NFDM_002 – Werte der Zugriffsprotokolleinträge auf der eGK“</td></tr> <tr> <td>accessType</td><td>gemäß Tabelle „Tab_FLA_NFDM_002 – Werte der Zugriffsprotokolleinträge auf der eGK“</td></tr> </table> Beschreibung <p>Ist die ermittelte Version eGK <code>cardVersion</code> G2.1 oder höher dann erfolgt das Schreiben des Zugriffsprotokolleintrags, andernfalls wird kein Protokolleintrag geschrieben (erfolgt durch KTR-AdV). Ein Zugriffsprotokolleintrag wird als Record in die Datei <code>EF.Logging</code> auf der eGK geschrieben. Tritt bei der Zugriffsprotokollierung ein Fehler auf, bricht die Operation mit Fehler 108 ab.</p>	cardVersion	In 1 für eGK ermittelte Version	dataType	gemäß Tabelle „Tab_FLA_NFDM_002 – Werte der Zugriffsprotokolleinträge auf der eGK“	accessType	gemäß Tabelle „Tab_FLA_NFDM_002 – Werte der Zugriffsprotokolleinträge auf der eGK“		
cardVersion	In 1 für eGK ermittelte Version								
dataType	gemäß Tabelle „Tab_FLA_NFDM_002 – Werte der Zugriffsprotokolleinträge auf der eGK“								
accessType	gemäß Tabelle „Tab_FLA_NFDM_002 – Werte der Zugriffsprotokolleinträge auf der eGK“								
3	Version der DPE-Speicherstruktur der eGK prüfen								
3.1	Aufruf Plattformbaustein PL_TUC_CARD_READ_FILE <table> <tr> <td colspan="2">Eingangsdaten</td></tr> <tr> <td>IDENTIFIKATOR</td><td><code>EF.StatusDPE</code></td></tr> <tr> <td colspan="2">Beschreibung</td></tr> <tr> <td colspan="2">Mittels PL_TUC_CARD_READ_FILE wird das Informationselement <code>Version_Speicherstruktur</code> der Datei <code>EF.StatusDPE</code> der eGK ausgelesen.</td></tr> </table>	Eingangsdaten		IDENTIFIKATOR	<code>EF.StatusDPE</code>	Beschreibung		Mittels PL_TUC_CARD_READ_FILE wird das Informationselement <code>Version_Speicherstruktur</code> der Datei <code>EF.StatusDPE</code> der eGK ausgelesen.	
Eingangsdaten									
IDENTIFIKATOR	<code>EF.StatusDPE</code>								
Beschreibung									
Mittels PL_TUC_CARD_READ_FILE wird das Informationselement <code>Version_Speicherstruktur</code> der Datei <code>EF.StatusDPE</code> der eGK ausgelesen.									
3.2	NFDM:checkContainerVersion <table> <tr> <td colspan="2">Eingangsdaten</td></tr> <tr> <td colspan="2">In 3.1 gelesener Wert des Informationselements <code>Version_Speicherstruktur</code> der Datei <code>EF.StatusDPE</code> der eGK.</td></tr> <tr> <td colspan="2">Beschreibung</td></tr> <tr> <td colspan="2">Es wird überprüft, ob FLA-NFDM die in 3.1 gelesene Versionsnummer der DPE-Speicherstruktur der eGK bekannt ist. Ist dies nicht der Fall, bricht die Operation mit Fehler 5104 ab.</td></tr> </table>	Eingangsdaten		In 3.1 gelesener Wert des Informationselements <code>Version_Speicherstruktur</code> der Datei <code>EF.StatusDPE</code> der eGK.		Beschreibung		Es wird überprüft, ob FLA-NFDM die in 3.1 gelesene Versionsnummer der DPE-Speicherstruktur der eGK bekannt ist. Ist dies nicht der Fall, bricht die Operation mit Fehler 5104 ab.	
Eingangsdaten									
In 3.1 gelesener Wert des Informationselements <code>Version_Speicherstruktur</code> der Datei <code>EF.StatusDPE</code> der eGK.									
Beschreibung									
Es wird überprüft, ob FLA-NFDM die in 3.1 gelesene Versionsnummer der DPE-Speicherstruktur der eGK bekannt ist. Ist dies nicht der Fall, bricht die Operation mit Fehler 5104 ab.									
4	DPE-Status-Flag auf eGK setzen								

	Aufruf Plattformbaustein PL_TUC_CARD_UPDATE_FILE	
	Eingangsdaten	
	IDENTIFIKATOR	EF.StatusDPE
	OFFSET	0
	NEWDATA	1
	Beschreibung	
	<p>Zur „Transaktionssicherung“ wird der Wert des Informationselements <code>Status</code> der Datei <code>EF.StatusDPE</code> der eGK vor Beginn des Löschvorgangs des DPE auf „1“ gesetzt.</p> <p>Liefert PL_TUC_CARD_UPDATE_FILE als Ergebnis <code>`MemoryFailure`</code>, bricht die Operation mit Fehler 5000 ab.</p> <p>In allen anderen Fehlerfällen bricht die Operation mit Fehler 5112 ab.</p>	
5	Größenangabe DPE auf eGK auf 0 setzen	
	Das Setzen der Größenangabe für den DPE auf der eGK auf 0 erfolgt implizit in 6.	
6	DPE von eGK löschen	
	Plattformbaustein PL_TUC_CARD_ERASE_FILE	
	Eingangsdaten	
	IDENTIFIKATOR	EF.DPE
	Beschreibung	
	<p>Der Inhalt der Datei <code>EF.DPE</code> auf der eGK wird durch Oktette mit dem Wert <code>'00'</code> (NULL) überschrieben. Dadurch wird einerseits der Inhalt des Informationselements <code>Länge DPE</code> mit <code>'00 00'</code> überschrieben und zudem die Daten des im Informationselement <code>DPE</code> gespeicherten DPE „gelöscht“, d. h. mit <code>'00'</code> (NULL) überschrieben.</p> <p>Gibt PL_TUC_CARD_ERASE_FILE den Fehlercode <code>MemoryFailure</code> der eGK zurück, bricht die Operation mit Fehler 5000 ab.</p> <p>In allen anderen Fehlerfällen bricht die Operation mit Fehler 5112 ab.</p>	
7	DPE-Zeitstempel auf eGK aktualisieren	
7.1	NFDM:getTime	
	Beschreibung	
	Liefere Systemzeit, die durch den Plattformbaustein PL_TUC_NET_SYNC_TIME	

	bereitgestellt wird.	
7.2	NFDM:formatSystemTime	
	Eingangsdaten	
	Im vorherigen Teilschritt ermittelte Systemzeit	
	Beschreibung	
	<p>Die Systemzeit wird für das Schreiben in das Informationselement <code>Timestamp</code> der Datei <code>EF.StatusDPE</code> der eGK gemäß [gemSpec_eGK_Fach_NFDM#3.2] formatiert (14 Oktette; YYYYMMDDhhmmss).</p> <p>Das eigentliche Schreiben des DPE-Zeitstempels in das Informationselement <code>Timestamp</code> der Datei <code>EF.StatusDPE</code> der eGK erfolgt aus Performancegründen in 9.2. Dort wird die Oktettkette zur Befüllung der Datei <code>EF.StatusDPE</code> als Ganzes in die Datei <code>EF.StatusDPE</code> der eGK geschrieben.</p> <p>Tritt hierbei ein Fehler auf, bricht die Operation mit Fehler 5112 ab.</p>	
8	Versionsnummer DPE-Speicherstruktur der eGK auf eGK aktualisieren	
	<p>Das Aktualisieren der Versionsnummer der DPE-Speicherstruktur der eGK erfolgt implizit im nächsten Schritt. Dort wird die Oktettkette zur Befüllung des DPE-Status-Containers als Ganzes in die Datei <code>EF.StatusDPE</code> der eGK geschrieben.</p>	
9	DPE-Status-Flag auf eGK zurücksetzen	
9.1	NFDM:concatenate	
	Eingangsdaten	
	In 7.2 formatierter DPE-Zeitstempel, Versionsnummer der DPE-Speicherstruktur der eGK, gemäß derer der DPE auf die eGK geschrieben wird.	
	Beschreibung	
	<p>Die Oktettkette für den DPE-Status-Container wird gemäß [gemSpec_eGK_Fach_NFDM#3.2] konkateniert, wobei das Informationselement <code>Version_XML</code> mit 5 Oktetten des Wertes '00' zu befüllen ist und das Informationselement <code>Status</code> den Wert „0“ erhält.</p> <p>Tritt hierbei ein Fehler auf, bricht die Operation mit Fehler 5112 ab.</p>	
9.2	Plattformbaustein PL_TUC_CARD_UPDATE_FILE	
	Eingangsdaten	
	IDENTIFIKATOR	EF.StatusDPE

OFFSET	0
NEWDATA	0
Beschreibung Mit Beendigung des Löschvorgangs des DPE auf die eGK wird das DPE-Status-Flag (Wert des Informationselements <code>Status</code> der Datei <code>EF.StatusDPE</code> der eGK) wieder zurück auf „0“ gesetzt und der DPE-Zeitstempel sowie die Versionsnummer der DPE-Speicherstruktur der eGK gemäß [gemSpec_eGK_Fach_NFDM#3.2] aktualisiert. Liefert <code>PL_TUC_CARD_UPDATE_FILE</code> als Ergebnis <code>`MemoryFailure`</code> , bricht die Operation mit Fehler 5000 ab. In allen anderen Fehlerfällen bricht die Operation mit Fehler 5112 ab.	

6.2 Schnittstelle I_COPY_DATA

FLA-NFDM stellt mit der Schnittstelle `I_COPY_DATA` Operationen in KTR-AdV zur Verfügung, die in NFDM das Funktionsmerkmal „Datenübertragung bei Kartentausch“ unterstützen. Der Datensatz „Persönliche Erklärungen“ soll hierüber im Rahmen der Bereitstellung einer neuen eGK des Versicherten von der alten Karte auf die neue Karte übertragen werden. Es handelt sich hierbei um eine interne Schnittstelle, deren Signatur und Verhalten definiert wird, jedoch keine Festlegungen hinsichtlich Implementierung getroffen werden.

Die Schnittstelle `I_COPY_DATA` umfasst die 2 Operationen:

- `GetData`
- `PutData`

6.2.1 Schnittstellendefinition

6.2.1.1 GetData

NFDM-A_2403 - Operation GetData

FLA-NFDM MUSS für KTR-AdV die Operation `GetData` des Funktionsmerkmals „Datenübertragung bei Kartentausch“ gemäß Tabelle „Tab_FLN_NFDM_014 – Operation `GetData`“ anbieten.

Tabelle 14: Tab_FLN_NFDM_014 – Operation GetData

Name	<code>GetData</code>
Beschreibung	Die Operation liest den im Aufrufparameter <code>application</code> benannten Datensatz DPE von der eGK und gibt ihn über den Parameter <code>Document</code> an den Aufrufer zurück.
Vorbedingung	In KTR-AdV werden vor Aufruf dieser Operation die übergreifenden Vorbedingungen aus Tabelle „Tab_FLN_NFDM_004 – Übergreifende Vorbedingungen für FLA-NFDM“ geprüft.

Erfolgsbedingungen	Die Operation MUSS die folgenden Erfolgsbedingungen überprüfen.	
	ID	Bedingung
	E1	Der auf der eGK gespeicherte Datensatz ist technisch konsistent, d. h. der Wert des Informationselements <i>Status</i> des Datensatzes in der Datei <i>EF.StatusDPE</i> der eGK ist „0“.
	E2	Die Version der internen Speicherstruktur (s. 6.3.1) der auf der eGK wird von FLA-NFDM unterstützt.
	E3	Es ist ein gemäß [RFC1952] gzip-komprimierter DPE auf der eGK gemäß [gemSpec_eGK_Fach_NFDM#2.1] gespeichert, d. h. das Informationselement <i>Länge</i> <i>DPE</i> der Datei <i>EF.DPE</i> der eGK hat einen Wert ungleich '00 00'.
	E4	Der auf der eGK gespeicherte Datensatz DPE ist valide gegen das entsprechende XML-Schema für den Datensatz DPE (s. [gemSpec_InfoNFDM#5]).
Aufrufparameter	Name	Beschreibung
	<i>objsysVersion</i>	Version des Objektsystems der eGK, von der der Datensatz DPE gelesen werden soll
	<i>application</i>	Identifiziert die eGK für den Datensatz, dessen Daten gelesen werden sollen. Folgende Werte sind zulässig: <i>DF.DPE</i>
Rückgabe	Name	Beschreibung
	<i>document</i>	Von der eGK des Versicherten gelesener, dekomprimierter Datensatz DPE, der durch Aufrufparameter <i>application</i> benannt wurde. Folgende XML-Dokumente sind zulässig: <i>DPE_Document</i> gemäß <i>gemSpec_KTR-AdV</i>
Nachbedingungen	Falls alle Erfolgsbedingungen erfüllt sind, MUSS die Operation die folgenden Nachbedingungen erfüllen.	
	ID	Bedingung
	N1	Der Datensatz DPE (Ausgabeparameter <i>document</i>) ist dekomprimiert.

Ablauf	<p>Der Ablauf der Operation <code>GetData</code>, ist im Aktivitätsdiagramm der Abbildung „Abbildung 4: Abb_FLA_NFDM_004 – Ablauf <code>GetData</code>“ modelliert. Die Umsetzung des Ablaufs spezifiziert Tabelle „Tabelle 16: Tab_FLA_NFDM_016 – Umsetzung Ablaufaktivitäten <code>GetData</code>“. Der Hersteller kann von der Umsetzung bzw. den spezifizierten Abläufen (z. B. zum Zwecke der Performanceoptimierung) abweichen, falls dadurch das definierte Außenverhalten der Operation gewährleistet bleibt.</p>	
Fehlermeldungen	<p>Für die generischen Fehlermeldungen finden sich die Attribute <code>ErrorType</code>, <code>Severity</code>, <code>Fehlertext</code>, <code>Befüllung Details</code> und <code>Auslösende Bedingung</code> in [gemSpec_OM#3.2.2]. Für die spezifischen Fehlermeldungen sind die Attribute <code>ErrorType</code>, <code>Severity</code>, <code>Fehlertext</code> und <code>Befüllung Details</code> in Tabelle „Tab_FLA_NFDM_018 – Fehlermeldungen“ definiert.</p> <p>Generische Fehlermeldungen</p>	
	Code	Befüllung Details
	108	Der Detailtext KANN den Fehler näher beschreiben.
	111	Der Detailtext KANN den Fehler näher beschreiben.
	Code	Befüllung Details
	5103	E1 ist nicht erfüllt.
	5104	E2 ist nicht erfüllt.
	5106	Die Dekomprimierung des DPE ist gescheitert.
	5114	E4 ist nicht erfüllt.
	5121	E3 ist nicht erfüllt.
	5500	Jegliches fehlerhafte Verhalten, das nicht durch die anderen Fehlermeldungen erfasst wird.

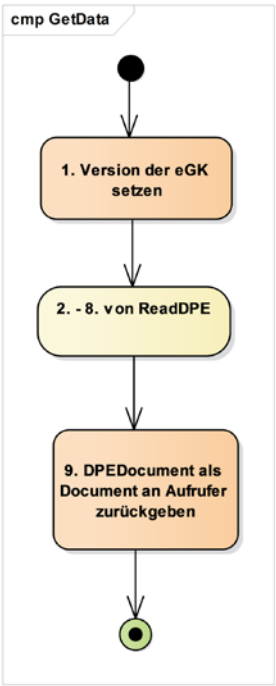


Abbildung 4: Abb_FLA_NFDM_004 – Ablauf GetData

[<=]

6.2.1.2 PutData

NFDM-A_2404 - Operation PutData

FLA-NFDM MUSS für KTR-AdV die Operation PutData des Funktionsmerkmals „Datenübertragung bei Kartentausch“ gemäß Tabelle „Tab_FLA_NFDM_015 – Operation PutData“ anbieten.

Tabelle 15: Tab_FLA_NFDM_015 – Operation PutData

Name	PutData	
Beschreibung	Die Operation schreibt im Aufrufparameter application benannten Datensatz DPE_Document gemäß gemSpec_KTR-AdV in das Informationselement DPE der Datei EF.DPE.	
Vorbedingung	In KTR-AdV werden vor Aufruf dieser Operation die übergreifenden Vorbedingungen aus Tabelle „Tab_FLA_NFDM_004 – Übergreifende Vorbedingungen für FLA-NFDM“ und folgende Vorbedingung geprüft:	
	ID	Bedingung
	V1	Das Zertifikat C.CH.AUTN der eGK ist zeitlich gültig und nicht gesperrt.

	V2	Die KVNR der eGK, von der der DPE Datensatz stammt (Document) ist gleich der KVNR der eGK für welche die Operation PutData aufgerufen wird (Die ersten zehn unveränderbaren Stellen der Krankenversichertennummer (KVNR) werden für den Vergleich herangezogen.).
Erfolgsbedingungen	Die Operation MUSS die folgenden Erfolgsbedingungen überprüfen.	
	ID	Bedingung
	E1	Der übergebene DPE ist valide gegen das XML-Schema für den DPE (s. [gemSpec_InfoNFDM#5]).
	E2	Der Versicherte der eGK ist mit dem Versicherten des DPE identisch, d. h. die Versicherten-ID (die ersten zehn unveränderbaren Stellen der Krankenversichertennummer (KVNR)) im Feld <code>organizationalUnitName</code> des Subject Distinguished Name (SubjectDN) des Authentisierungszertifikats der eGK (gespeichert in der Datei <code>EF.C.CH.AUTN.R2048</code>) ist die gleiche wie die im DPE im Element <code>Versicherten_ID</code> gespeicherte. <i>Anmerkung: Es gibt zwei <code>organizationalUnitName</code>-Felder im SubjectDN. Das zehnstellige, alphanumerische Feld beinhaltet die Versicherten-ID (unveränderbarer Teil der Krankenversichertennummer), das andere, neunstellige numerische Feld das Institutionskennzeichen der Krankenversichertennummer (s. [gemSpec_PKI#5.1.3.1]).</i>
	E3	Der DPE ist nicht größer als der auf der eGK im Informationselement <code>DPE</code> der Datei <code>EF.DPE</code> der eGK zur Verfügung stehende Speicherplatz.
	E4	Es ist auf der eGK kein DPE bereits vorhanden, d. h. das Informationselement <code>Länge DPE</code> der Datei <code>EF.DPE</code> der eGK hat einen Wert gleich '00 00'.
Aufrufparameter	Name	Beschreibung
	<code>objsysVersionSrc</code>	Version des Objektsystems der eGK, von der der Datensatz DPE ursprünglich gelesen werden soll
	<code>objsysVersionDest</code>	Version des Objektsystems der eGK, auf die der Datensatz DPE geschrieben werden soll
	<code>application</code>	Identifiziert die eGK für den Datensatz, dessen Daten geschrieben werden sollen. Folgende Werte sind zulässig: <code>DF.DPE</code>

	document	Der durch Aufrufparameter <code>application</code> benannte und auf die eGK zu schreibende Datensatz DPE. Folgende XML-Dokumente sind zulässig: <code>DPE_Document</code> gemäß <code>gemSpec_KTR-AdV</code>
Rückgabe	Name	Beschreibung
Nachbedingungen	Falls alle Erfolgsbedingungen erfüllt sind, MUSS die Operation die folgenden Nachbedingungen erfüllen.	
	ID	Bedingung
	N1	Der im Aufrufparameter <code>Document</code> übergebene Datensatz DPE ist gemäß [RFC1952] gzip-komprimiert auf der eGK im Informationselement <code>DPE</code> der Datei <code>EF.DPE</code> gemäß [gemSpec_eGK_Fach_NFDM#3.1] gespeichert.
	N2	Die Größe des DPE in Oktett ist im Informationselement <code>Länge DPE</code> der Datei <code>EF.DPE</code> der eGK gemäß [gemSpec_eGK_Fach_NFDM#3.1] gespeichert.
	N3	Der Wert des Informationselements <code>Timestamp</code> der Datei <code>EF.StatusDPE</code> der eGK ist aktualisiert.
	N4	Der Wert des Informationselements <code>Version_Speicherstruktur</code> der Datei <code>EF.StatusDPE</code> der eGK ist aktualisiert mit einer gemäß [gemSpec_eGK_Fach_NFDM#3.2] gültigen Versionsnummer.
	N5	Der Wert des Informationselement <code>Status</code> der Datei <code>EF.StatusDPE</code> der eGK ist „0“.
Ablauf	Der Ablauf der Operation <code>PutData</code> , ist im Aktivitätsdiagramm der Abbildung „Abb_FLA_NFDM_005 – Ablauf PutData“ modelliert. Die Umsetzung des Ablaufs spezifiziert Tabelle „Tabelle 17: Tab_FLA_NFDM_017 – Umsetzung Ablaufaktivitäten <code>PutData</code> “. Der Hersteller kann von der Umsetzung bzw. den spezifizierten Abläufen (z. B. zum Zwecke der Performanceoptimierung) abweichen, falls dadurch das definierte Außenverhalten der Operation gewährleistet bleibt.	
Fehlermeldungen	Für die generischen Fehlermeldungen finden sich die Attribute <code>ErrorType</code> , <code>Severity</code> , <code>Fehlertext</code> , <code>Befüllung Details</code> und <code>Auslösende Bedingung</code> in [gemSpec_OM#3.2.2]. Für die spezifischen Fehlermeldungen sind die Attribute <code>ErrorType</code> , <code>Severity</code> , <code>Fehlertext</code> und <code>Befüllung Details</code> in Tabelle „Tab_FLA_NFDM_018 – Fehlermeldungen“ definiert. Generische Fehlermeldungen	

Code	Befüllung Details
108	Der Detailtext KANN den Fehler näher beschreiben.
112	Der Detailtext KANN den Fehler näher beschreiben.
Spezifische Fehlermeldungen	
Code	Auslösende Bedingung
5000	Die eGK ist defekt.
5104	Die Version der internen DPE-Speicherstruktur wird von FLA-NFDM nicht unterstützt. Eine Prüfung der Existenz eines DPE kann nicht erfolgen.
5108	E2 ist nicht erfüllt.
5110	Die Komprimierung des DPE ist gescheitert.
5113	E3 ist nicht erfüllt
5114	E1 nicht erfüllt.
5122	E4 ist nicht erfüllt.
5500	Jegliches fehlerhafte Verhalten, das nicht durch die anderen Fehlermeldungen erfasst wird.

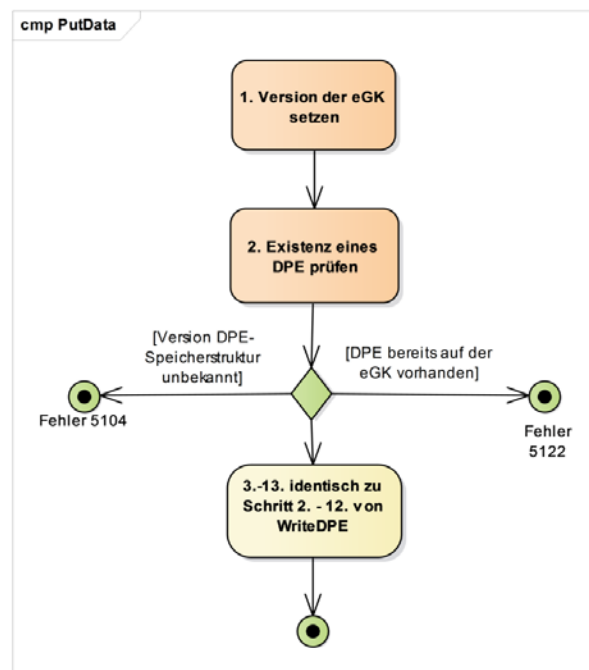


Abbildung 5: Abb_FLA_NFDM_005 – Ablauf PutData

[<=]

6.2.2 Umsetzung

Die folgenden Unterkapitel beschreiben die Umsetzung der Operationsabläufe von I_COPY_DATA in der KTR-AdV Umgebung.

6.2.2.1 GetData

Tabelle 16: Tab_FLA_NFDM_016 – Umsetzung Ablaufaktivitäten GetData

1	Version der eGK setzen
	In <code>CardVersion</code> wird die Information aus dem Aufrufparameter <code>objsysVersion</code> übernommen.
2-8	Identisch zu 2-8 von ReadDPE (s. Tabelle „Tabelle 11: Tab_FLA_NFDM_010 – Umsetzung Ablaufaktivitäten ReadDPE“)
9	DPEDocument als Document an Aufrufer zurückgeben
	Rückgabe <code>Document</code>

6.2.2.2 PutData

Tabelle 17: Tab_FLA_NFDM_017 – Umsetzung Ablaufaktivitäten PutData

1	Version der eGK setzen		
	In <code>CardVersion</code> wird die Information aus dem Aufrufparameter <code>objsysVersionDest</code> übernommen.		
2	Existenz eines DPE prüfen		
2.1	Aufruf Plattformbaustein <code>PL_TUC_CARD_READ_FILE</code>		
	Eingangsdaten		
	<table> <tr> <td>IDENTIFIKATOR</td><td>EF.StatusDPE</td></tr> </table>	IDENTIFIKATOR	EF.StatusDPE
IDENTIFIKATOR	EF.StatusDPE		
	Beschreibung		
	Es werden alle Informationselemente der Datei <code>EF.StatusDPE</code> der eGK gelesen.		
2.2	<code>NFDM:checkConsistency</code>		
	Eingangsdaten		
	In 2.1 aus der Datei <code>EF.StatusPDE</code> gelesene Daten (= alle Informationselemente)		
	Beschreibung		

	<p>Ist der Wert des Informationselements <code>Status</code> der Eingangsdaten „0“, dann ist der DPE technisch konsistent. Ist der Wert „1“, ist der DPE technisch inkonsistent und kann auch dann überschrieben werden, wenn auf der eGK bereits ein DPE angelegt ist. Ist der Wert weder „0“ noch „1“ gemäß [gemSpec_eGK_Fach_NFDM#2.2], so wurde noch kein DPE angelegt.</p>	
2.3	NFDM:checkContainerVersion	
	Eingangsdaten	
	In 2.1 aus dem DPE-Statuscontainer der eGK gelesene Daten (= alle Informationselemente)	
	Beschreibung	
	Es wird überprüft, ob FLA-NFDM die im Informationselement <code>Version_Speicherstruktur</code> der Eingangsdaten gespeicherte Versionsnummer der DPE-Speicherstruktur der eGK bekannt ist. Ist dies nicht der Fall, bricht die Operation mit Fehler 5104 ab.	
2.4	Aufruf Plattformbaustein PL_TUC_CARD_READ_FILE	
	Eingangsdaten	
	IDENTIFIKATOR	EF.DPE
	LENGTH	‘02’
	Beschreibung	
	Es wird das Informationselement <code>Länge DPE</code> der Datei <code>EF.DPE</code> der eGK gelesen. Tritt dabei ein Fehler auf, bricht die Operation mit Fehler 111 ab.	
2.5	NFDM:checkSize	
	Eingangsdaten	
	In 2.1 gelesene Daten aus der Datei <code>EF.DPE</code> .	
	Beschreibung	
	Es wird das Informationselement <code>Länge DPE</code> der Eingangsdaten als Größe des DPE (Anzahl Oktette) ausgewertet. Ist die Größe ungleich ‘00 00’, bedeutet dies, es existiert ein DPE auf der eGK. Existiert ein DPE auf der eGK und ist der Datensatz entsprechend der Prüfung in 2.2 konsistent, so bricht die Operation mit Fehler 5122 ab.	
3-13	Identisch zu 2-12 von <code>WriteDPE</code> (s. Tabelle „Tabelle 12: Tab_FLA_NFDM_011 – Umsetzung Ablaufaktivitäten <code>WriteDPE</code> “), wobei der Aufrufparameter <code>Document</code> als <code>DPEDocument</code> verarbeitet wird.	

6.3 Artefakte

6.3.1 DPE-Speicherstruktur der eGK

Die DPE-Speicherstruktur der eGK wird in [gemSpec_eGK_Fach_NFDM] definiert. Die Version der DPE-Speicherstruktur wird im Informationselement `Version_Speicherstruktur` der Datei `EF.StatusDPE` der eGK gespeichert.

6.3.2 Der DPE auf der eGK

Der DPE ist auf der Karte in der Datei `EF.DPE` gemäß [RFC1952] gzip-komprimiert gespeichert. Die XML-Struktur des DPE ist in [gemSpec_InfoNFDM#5] definiert.

6.4 Testunterstützung

Zur Unterstützung von Tests im Zusammenhang mit den Funktionsmerkmalen werden keine gesonderten Festlegungen getroffen.

6.5 Hardwaremerkmale

Die Funktionsmerkmale setzen keine besonderen Hardwaremerkmale voraus.

7 Informationsmodell

Das (technische) Informationsmodell NFDM ist in [gemSpec_InfoNFDM] spezifiziert.
Schnittstellenbeschreibungen und Speicherstruktur der eGK sind im jeweiligen
Unterkapitel „Artefakte“ der Funktionsmerkmale aufgeführt.

8 Verteilungssicht

FLA-NFDM ist integraler Bestandteil von KTR-AdV. Die Hardware-Merkmale von KTR-AdV sind in [gemSpec_KTR-AdV] spezifiziert.

Eine weitergehende Darstellung der hardwareseitigen Verteilung von FLA-NFDM bzw. seiner Teilsysteme und der Einbettung in die physikalische Umgebung wird daher nicht benötigt.

9 Anhang A – Verzeichnisse

9.1 – Abkürzungen

Kürzel	Erläuterung
AID	Application Identifier
AdV	Anwendungen des Versicherten
C2C	Card-to-Card
DF	Dedicated File
DPE	Datensatz „Persönliche Erklärungen“
EF	Elementary File
eGK	elektronische Gesundheitskarte
FID	File Identifier
FLA-NFDM	Fachlogik der Fachanwendung Notfalldatenmanagement auf Seite der AdV-App
ID	Identifier
KVNR	Krankenversichertennummer
NFD	Notfalldatensatz
NFDM	Notfalldaten-Management
PID	Password Identifier
PIN	Personal Identification Number
RFC	Request for Comments
SM-B	Security Module Card
SubjectDN	Subject Distinguished Name
TI	Telematikinfrastruktur

9.2 – Glossar

Das Glossar wird als eigenständiges Dokument [gemGlossar] zur Verfügung gestellt.

9.3 – Abbildungsverzeichnis

Abbildung 1: Abb_FLA_NFDM_001 – Ablauf ReadDPE	19
Abbildung 2: Abb_FLA_NFDM_002 – Ablauf WriteDPE.....	22
Abbildung 3: Abb_FLA_NFDM_003 – Ablauf EraseDPE.....	25
Abbildung 4: Abb_FLA_NFDM_004 – Ablauf GetData	41
Abbildung 5: Abb_FLA_NFDM_005 – Ablauf PutData	44

9.4 – Tabellenverzeichnis

Tabelle 1: Tab_FLA_NFDM_001 Verwendete Bezeichner für Objekte der eGK.....	6
Tabelle 2: Tab_FLA_NFDM_019 Abbildung NFDM-Anwendungsfälle auf Operationen des FLA-NFDM	8
Tabelle 3: Tab_FLA_NFDM_002 – Werte der Zugriffsprotokolleinträge auf der eGK	13
Tabelle 4: Tab_FLA_NFDM_003 – Terminal-Anzeigen für PIN-Eingabe.....	13
Tabelle 5: Tab_FLA_NFDM_004 – Übergreifende Vorbedingungen für FLA-NFDM	14
Tabelle 6: Tab_FLA_NFDM_005 – Übergreifende Erfolgsbedingungen.....	15
Tabelle 7: Tab_FLA_NFDM_006 – Übergreifende Nachbedingungen	15
Tabelle 8: Tab_FLA_NFDM_007 – Operation ReadDPE	16
Tabelle 9: Tab_FLA_NFDM_008 – Operation WriteDPE	19
Tabelle 10: Tab_FLA_NFDM_009 – Operation EraseDPE	23
Tabelle 11: Tab_FLA_NFDM_010 – Umsetzung Ablaufaktivitäten ReadDPE	26
Tabelle 12: Tab_FLA_NFDM_011 – Umsetzung Ablaufaktivitäten WriteDPE	29
Tabelle 13: Tab_FLA_NFDM_012 – Umsetzung Ablaufaktivitäten EraseDPE	34
Tabelle 14: Tab_FLA_NFDM_014 – Operation GetData.....	38
Tabelle 15: Tab_FLA_NFDM_015 – Operation PutData	41
Tabelle 16: Tab_FLA_NFDM_016 – Umsetzung Ablaufaktivitäten GetData.....	45
Tabelle 17: Tab_FLA_NFDM_017 – Umsetzung Ablaufaktivitäten PutData	45
Tabelle 18: Tab_FLA_NFDM_018 – Fehlermeldungen FLA-NFDM	54

9.5 – Referenzierte Dokumente

9.5.1 – Dokumente der gematik

Die nachfolgende Tabelle enthält die Bezeichnung der in dem vorliegenden Dokument referenzierten Dokumente der gematik zur Telematikinfrastruktur. Der mit der vorliegenden Version korrelierende Entwicklungsstand dieser Konzepte und Spezifikationen wird pro Release in einer Dokumentenlandkarte definiert, Version und Stand der referenzierten Dokumente sind daher in der nachfolgenden Tabelle nicht aufgeführt. Deren zu diesem Dokument passende jeweils gültige Versionsnummer sind in der aktuellsten, von der gematik veröffentlichten Dokumentenlandkarte enthalten, in der die vorliegende Version aufgeführt wird.

[Quelle]	Herausgeber: Titel
[DPE_Document.xsd]	gematik: XML-Schema-Dokument für den DPE
[gemGlossar]	gematik: Glossar der Telematikinfrastruktur
[gemLH_NFDM]	Projektteam NFDM: Lastenheft Notfalldaten-Management
[gemSysL_AdV]	gematik: Systemspezifisches Konzept Anwendungen des Versicherten (AdV)
[gemSysL_NFDM]	gematik: Systemspezifisches Konzept Notfalldaten-Management (NFDM)
[gemSpec_Karten_Fach_TIP]	gematik: Befüllvorschriften für die Plattformanteile der Karten der TI
[gemSpec_eGK_Fach_NFDM]	gematik: Speicherstrukturen der eGK für die Fachanwendung NFDM
[gemSpec_eGK_ObjSys]	gematik: Spezifikation der elektronischen Gesundheitskarte eGK-Objektsystem
[gemSpec_InfoNFDM]	gematik: Informationsmodell Notfalldaten-Management (NFDM)
[gemSpec_OM]	gematik: Übergreifende Spezifikation Operations und Maintenance (Fehlermanagement, Versionierung, Monitoring)
[gemSpec_PKI]	gematik: Spezifikation PKI
[gemSpec_KTR-AdV]	gematik: Spezifikation KTR-AdV

9.5.2 – Weitere Dokumente

[Quelle]	Herausgeber (Erscheinungsdatum): Titel
----------	--

[RFC1952]	RFC 1952 (Mai 1996): GZIP file format specification version 4.3, http://tools.ietf.org/html/rfc1952 (zuletzt geprüft am 31.01.2014)
[RFC2119]	RFC 2119 (März 1997): Key words for use in RFCs to Indicate Requirement Levels S. Bradner, http://tools.ietf.org/html/rfc2109 (zuletzt geprüft am 06.06.2012)

10 Anhang B

10.1 – Fehlermeldungen

Die Tabelle richtet sich nach den Vorgaben von [gemSpec_OM#5.3.1.2].

Tabelle 18: Tab_FLA_NFDM_018 – Fehlermeldungen FLA-NFDM

Code	ErrorType	Severity	ErrorText	Befüllung Details
5000	Technical	FATAL	Die eGK ist defekt.	Der Detailtext KANN den Fehler näher beschreiben.
5103	Technical	ERROR	Datensatz „Persönliche Erklärungen“ nicht konsistent	Der Detailtext MUSS den Zeitstempel des DPE aus dem Informationselement <code>Timestamp der Datei EF.StatusDPE</code> der eGK enthalten.
5104	Technical	FATAL	Unbekannte Version der Speicherstruktur für den Datensatz „Persönliche Erklärungen“ auf der eGK	Der Detailtext MUSS die Versionsnummer der DPE-Speicherstruktur auf der eGK (s. Kapitel 6.3.1) enthalten.
5106	Technical	ERROR	Dekomprimierung des Datensatz „Persönliche Erklärungen“ gescheitert	Der Detailtext KANN den Fehler näher beschreiben.
5108	Security	ERROR	Die Versicherten-ID des Datensatz „Persönliche Erklärungen“ stimmt nicht mit der Versicherten-ID der eGK überein.	Der Detailtext KANN den Fehler näher beschreiben.
5110	Technical	ERROR	Die Komprimierung des Datensatz „Persönliche Erklärungen“ ist gescheitert.	Der Detailtext KANN den Fehler näher beschreiben.
5112	Technical	ERROR	Das Löschen des Datensatz „Persönliche Erklärungen“ ist gescheitert.	Der Detailtext KANN den Fehler näher beschreiben.
5113	Business	ERROR	Der Datensatz „Persönliche Erklärungen“ überschreitet die maximal zulässige Größe.	Der Detailtext KANN den Fehler näher beschreiben.

5114	Security	ERROR	Der Datensatz „Persönliche Erklärungen“ ist nicht valide.	Der Detailtext KANN den Fehler näher beschreiben.
5121	Business	ERROR	Es ist kein Datensatz „Persönliche Erklärungen“ auf der eGK gespeichert.	Der Detailtext KANN den Fehler näher beschreiben.
5122	Business	ERROR	Es ist bereits ein Datensatz „Persönliche Erklärungen“ auf der eGK gespeichert.	Der Detailtext KANN den Fehler näher beschreiben.
5500	Technical	FATAL	Interner Fehler	Der Detailtext KANN den Fehler näher beschreiben.