Meldeverfahren EAU

Elektronische Arbeitsunfähigkeitsbescheinigung

– Einheitliche Kernprüfung –

Betriebshandbuch

Autor: Verband der Ersatzkassen e. V. (vdek)

Datum : 17.09.2026

Dokumentenversion: 3.0.0

Inhalt

1. Änderungsübersicht 3

2. Allgemeines 4

2.1 Voraussetzungen 4

2.2 Lieferumfang 4

2.3 Kernprüfungskonvention 5

3. Einheitliche Schnittstellen 6

4. Einbindung der Kernprüfung 8

5. Parallelbetrieb zweier Schemaversionen 9

1. Änderungsübersicht

Version des Dokumentes	Datum	Kapitel	Änderung
1.0.0	31.05.2023	– alle –	Initialer Stand
2.0.0	18.09.2024	– alle –	Überarbeitung und neues Kapitel 5
2.0.1	20.09.2024	2.2	Lieferumfang aktualisiert
2.0.3	10.09.2025	2.1, 2.2	Umstellung auf Java 11 und JAXB 4
3.0.0	15.09.2026	2.2, 5	Änderung bei Paketierung; Aktualisierung des Parallelbetriebs

2. Allgemeines

Dieses Dokument ist die technische Anleitung für die Installation und den Betrieb der Komponente für die Kernprüfung im Verfahren EAU.

2.1 Voraussetzungen

Als Laufzeit- / Entwicklungsumgebung wird Java Runtime Environment (JRE) oder Java Developer Kit (JDK) 11vorausgesetzt. Weitere Komponenten oder Programme werden nicht benötigt.

2.2 Lieferumfang

Die Kernprüfung wird in drei unterschiedlichen Varianten bereitgestellt:

1. singlejar,
2. fatjar
3. fatjar-with-jaxb

In der *singlejar* Variante werden sämtliche Abhängigkeiten und die Kernprüfung selbst als separate jar-Dateien geliefert. Dies hat den Vorteil, dass Dopplungen von Klassen im Classpath beim Einsatz mehrerer (vdek-)Kernprüfungen vermieden werden können.

In der *fatjar* Variante werden alle vdek-eigenen Klassen in einer jar-Datei gebündelt, so dass nur noch die JAXB-spezifischen Bibliotheken einzeln eingebunden werden müssen.

In der fatjar-with-jaxb sind alle benötigten Bibliotheken, inkl. technischer Basis und der JAXB-Abhägigkeiten, in einer einzigen jar-Datei gebündelt. Diese Version ist also ohne weitere Abhängigkeiten alleine lauffähig.

Im Folgenden wird beispielhaft der Inhalte dieser Varianten aufgezeigt:

	singlejar	fatjar	fatjar-with-jaxb
KernpruefungEAU-03.00.00	ja	ja	ja
kernpruefung-api-1.0.0	separat	enthalten	enthalten
kernpruefung-xmlbasis-02.02.01.004	separat	enthalten	enthalten
istack-commons-runtime-4.1.2.	separat	separat	enthalten
jakarta.activation-api-2.1.4	separat	separat	enthalten
jaxb-core-4.0.9	separat	separat	enthalten

jaxb-runtime-4.0.9	separat	separat	enthalten

Folgende Komponenten werden in komprimierter Archivform (EAU.nnn.zip) ausgeliefert:

- Benutzer-Betriebshandbuch: Dieses Dokument
- Versionshistorie: Änderungsdokumentation
- Java-Doc: Entwicklerdokumentation Prüfmodul EAU
- Kernprüfprogramm: Alle java- und class-Dateien des Prüfmoduls
- Datensatzbeschreibung: aktueller Fehlerkatalog als Grundlage
- Freigabeerklärung: Erklärung über die erfolgte Qualitätssicherung

2.3 Kernprüfungskonvention

Alle benötigten Java-Sourcen, sowie deren Kompilate und das Java-API sind in folgenden Archiven enthalten:

- KernpruefungEAU-03.00.00.jar
- KernpruefungEAU-03.00.00-sources.jar
- KernpruefungEAU-03.00.00-javadoc.jar

Die Namenskonvention des Java-Archives setzt sich zusammen aus dem Namen **Kernpruefung**, dem Verfahrensnamen (in diesem Fall **EAU**) und aus der Versionsnummer (in diesem Fall **03.00.00**). Die ersten beiden Ziffern dieser Nummer entsprechen der Versionsnummer aus der Datensatzbeschreibung. Die letzte Ziffer ist für die Bereinigung von Fehlern vorgesehen. (Ziffer 00 = Urversion dieses Kernprüfprogramms für eine Version einer Datensatzbeschreibung).

3. Einheitliche Schnittstellen

Für alle einheitlichen Kernprüfungen wurden zwei Java-Interfaces entworfen, die von den einzelnen Kernprüfverfahren implementiert werden. Die beiden Schnittstellen sind im Java-Package **`kernpruefung`** zusammengefasst. Sie enthalten die Methoden für die Kernprüfung. Die Implementierung der Methoden wird in den unterschiedlichen Kernprüfverfahren realisiert.

Das Interface **`kernpruefung.Kernpruefung`** enthält die Methode **`pruefe(..)`**.

Als Übergabeparameter wird der vollständige Datensatz einer Meldung als XML-String übergeben. Für die Prüfung des Datensatzes sind im Kassenwahlrecht-Datenaustauschverfahren Angaben aus dem Vorlaufsatz nicht notwendig, deswegen wird der Parameter Vorlaufsatz nicht ausgewertet und kann als Null-Objekt (**`null`**) übergeben werden. Es werden die Prüfungen aus der Datensatzbeschreibung bzw. des Fehlerkataloges durchgeführt. Das Ergebnis der Prüfung ist ein Rueckgabe-Objekt.

Im Interface **`kernpruefung.Rueckgabe`** sind Methoden für die Abfrage des Ergebnisses enthalten. Die Methode **`getReturnCode()`** gibt den Returncode als int-Wert zurück.

Es wurden folgende Returncodes festgelegt:

Returncode	Bedeutung
0	Kernprüfung fehlerfrei
1	Kernprüfung enthält Hinweise
2	Kernprüfung enthält Fehler
4	Prüfung konnte nicht durchgeführt werden und wurde abgebrochen.

Mit der Methode **`getRueckgabeMeldung()`** werden die Fehlerbausteine als String-Array zurückgegeben. Die Anzahl der Fehler oder Hinweise ist auf neun beschränkt.

Die String-Darstellung des Rueckgabe-Objektes erhält man mit der Methode `toString()`. Der Aufbau des Strings wird in der folgenden Tabelle beschrieben:

Stelle		Inhalt
von	bis	
1	1	Return-Code
2	2	Anzahl der Fehlerbausteine (max. 9)
3	78	Fehlerbaustein 1 (falls vorhanden)
79	154	Fehlerbaustein 2 (falls vorhanden)
155	230	Fehlerbaustein 3 (falls vorhanden)
231	306	Fehlerbaustein 4 (falls vorhanden)
307	382	Fehlerbaustein 5 (falls vorhanden)
383	458	Fehlerbaustein 6 (falls vorhanden)
459	534	Fehlerbaustein 7 (falls vorhanden)
535	610	Fehlerbaustein 8 (falls vorhanden)
611	686	Fehlerbaustein 9 (falls vorhanden)

4. Einbindung der Kernprüfung

Um die Kernprüfung in ein Verfahren einzubinden, ist es nötig die konkrete Schnittstellenimplementierung zu instanziieren:

```
import com.vdek.agv.eau.kp.KernpruefungEAUImpl;
KernpruefungEAUImpl kernpruefung = new KernpruefungEAUImpl();
```

Über die erzeugte Instanz `kernpruefung` können nun die Prüfungen mittels der Methode `pruefe(..)` durchgeführt werden.

```
Rueckgabe ruckgabe = kernpruefung.pruefe(xmlString, null);
```

Die Ergebnisse werden an das Rückgabeobjekt gereicht und können nun ausgewertet werden.

Zusammengefasst an einem Beispiel könnte die Einbindung folgendermaßen durchgeführt werden:

```
import kernpruefung.Kernpruefung;
import kernpruefung.Rueckgabe;
import com.vdek.agv.eau.kp.KernpruefungEAUImpl;

public class BeispielPrueferEAU
{
      public void validiereDatensatz()
      {
            String datensatz    =
             "<?xml version="1.0" encoding="ISO-8859-1" standalone="yes"?>"
            + "<ns3: Anforderung_eAU_AG … ";

            Kernpruefung kp = new KernpruefungEAUImpl();

            Rueckgabe rueckgabe = kp.pruefe( datensatz, null );

            System.out.println( rueckgabe.getReturnCode() );
            System.out.println( rueckgabe.getRueckgabeMeldungen() );
            System.out.println( rueckgabe.toString() );
      }
}
```

5. Parallelbetrieb zweier Schemaversionen

Diese Kernprüfung unterstützt den gleichzeitigen Betrieb der letzten beiden aktuellen Schemaversionen. Sie kann also sowohl für die Schemaversion ab dem 01.01.2027 als auch für die davor gültige Schemaversion eingesetzt werden.

Dies erleichtert so hoffentlich den Betrieb während der Übergangsfrist, in der beide Schemaversionen erlaubt sind und auch den Betrieb bei der Umstellung.