

XGewerbeanzeige

- Hersteller-Workshop, 6. Oktober 2015 -

Achim Mohr
Statistisches Bundesamt, Wiesbaden
achim.mohr@destatis.de

Alf Steilen
Statistisches Bundesamt, Wiesbaden
alf.steilen@destatis.de

Agenda

- **Datenformate DatML/RAW, DatML/RES, DatML/SDF und DatML/SDFmeta**
- **Funktionsweise und Schnittstellen des GWA-Prüftools**
- **Generisch unterstützte Erzeugung von Gewerbemeldungen**

- 1. Datenformate DatML/RAW, DatML/RES, DatML/SDF und DatML/SDFmeta**
2. Funktionsweise und Schnittstellen des GWA-Prüftools
3. Generisch unterstützte Erzeugung von Gewerbemeldungen

Datenformat DatML/RAW

Datenformat DatML/RAW

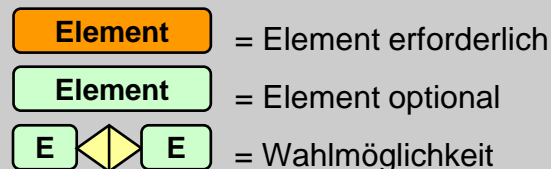
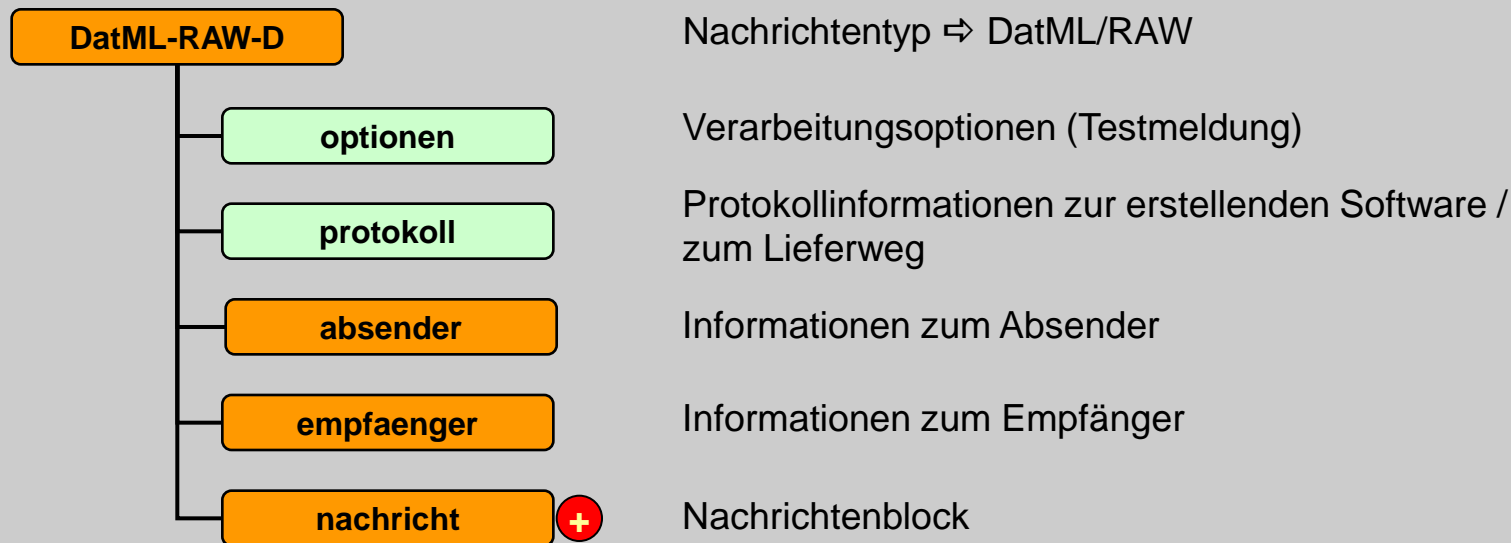
- **DatML/RAW ist ein XML-Format zur Lieferung von Meldungen (Rohdaten)**
 - dient der strukturierten Beschreibung von Rohdaten
- **Erhebungsübergreifend einsetzbar**
 - nur Anzahl und Art der Merkmale und Merkmalsgruppen sind erhebungsspezifisch
- **Mandantenfähig**
 - Versand und Empfang von Lieferungen kann durch Dritte erfolgen
- **Zusatzfunktionen**
 - Adresskorrekturen
 - Testmeldungen
- **Mehrmeldungsfähig**

Datenformat DatML/RAW

- **Ein DatML/RAW-Dokument ist hierarchisch gegliedert**
 - Dokumentebene
 - Angaben zu
 - Verarbeitung (Kennzeichnung von Testmeldung)
 - Protokollierung (Kennzeichnung der erstellenden Software, des Lieferweges)
 - Absender
 - Empfänger
 - Nachrichtenebene
 - Angaben zu
 - Erhebung (xGewerbeanzeige '0004')
 - Berichtszeitraum
 - Berichtspflichtigen
 - Berichtsempfänger
 - Materialnummer
 - Datenattribute (z.B. Dezimal- und Tausender-Trennzeichen)
 - Hilfsmerkmale (z.B. BerichtseinheitID oder Amtlicher Gemeindeschlüssel AGS)
 - Datenebene
 - Datensätze mit Merkmalen und Merkmalsgruppen

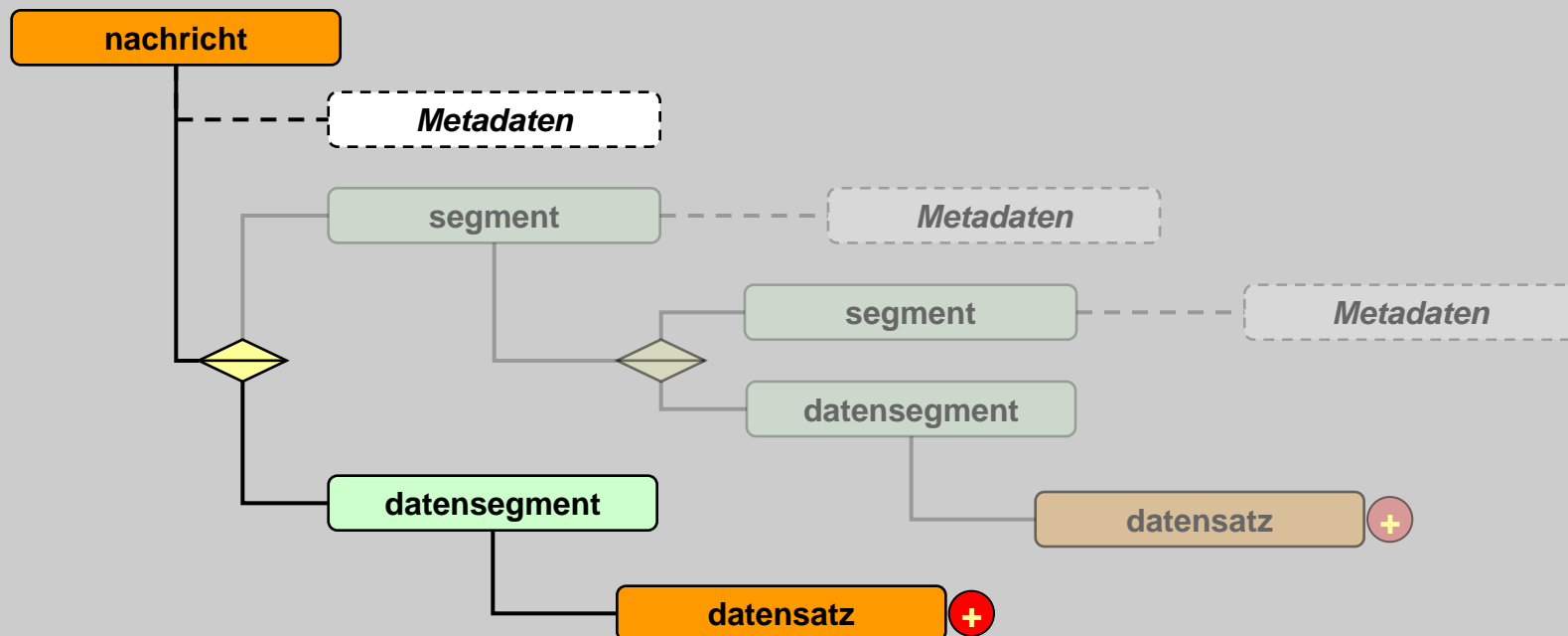
Datenformat DatML/RAW

■ Aufbau des DatML/RAW-Datenformates



Datenformat DatML/RAW

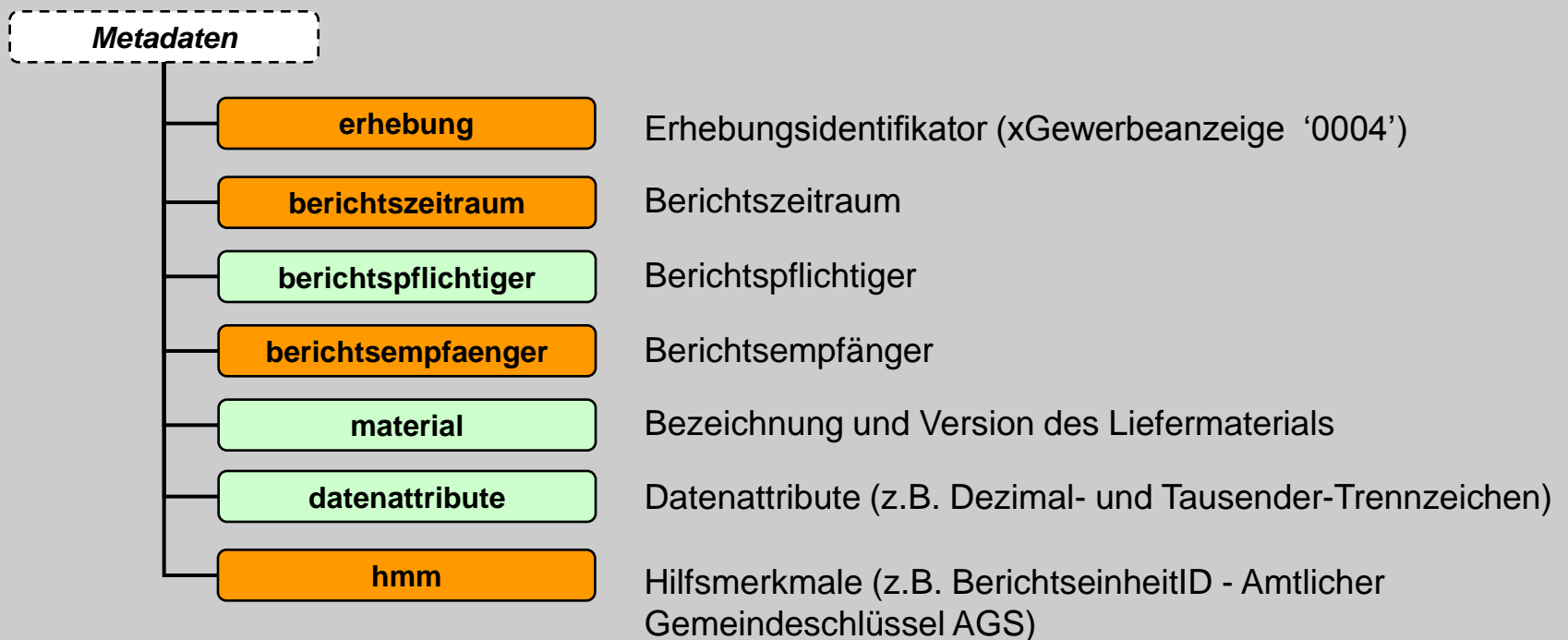
▪ Aufbau *<nachricht>*




- Element** = Element erforderlich
- Element** = Element optional
- E** **E** = Wahlmöglichkeit

Datenformat DatML/RAW

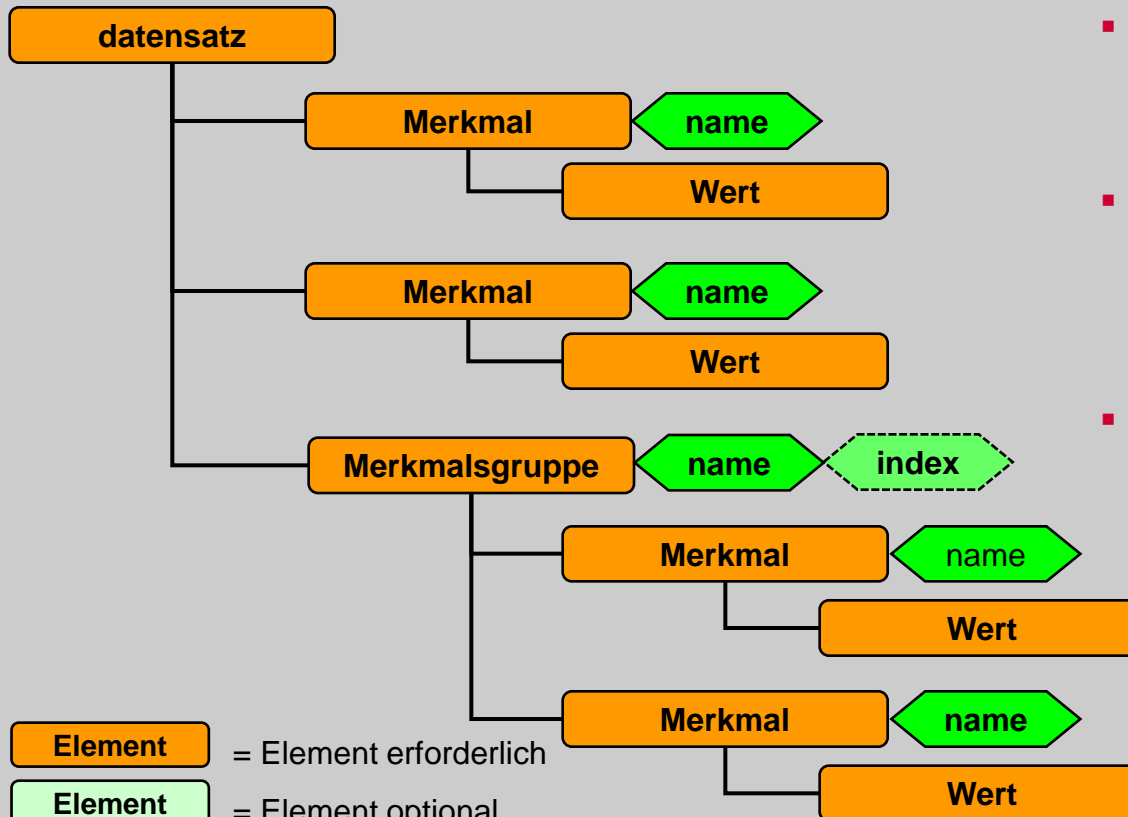
■ Aufbau <Metadaten>



- Element** = Element erforderlich
- Element** = Element optional
- E**  **E** = Wahlmöglichkeit

Datenformat DatML/RAW

▪ Aufbau *<datensatz>*



- Merkmalsnamen müssen innerhalb eines Datensatzes bzw. einer Merkmalsgruppe eindeutig sein.
- Es darf mehr als eine Merkmalsgruppe mit dem gleichen Namen geben ⇒ Index zur Merkmalsgruppe erforderlich.
- Ein Index kann direkt angegeben oder automatisch gebildet werden.

Element (orange rectangle) = Element erforderlich

Element (green rectangle) = Element optional

E (green rectangle) **E** (green rectangle) = Wahlmöglichkeit

Datenformat DatML/RES

Datenformat DatML/RES

- **DatML/RES ist ein XML-Format für Prüfprotokolle zu geprüften DatML/RAW-Dokumenten**
 - dient der strukturierten Beschreibung des Prüfprotokolls
- **Bildet die Struktur des geprüften DatML/RAW Dokumentes nach**
- **Anstelle von Daten entspricht die hierarchische Gliederung des Dokumentes den Ergebnissen der Prüfung der jeweiligen Ebene**
- **Wird für Gewerbemeldungen mittels des GWA-Prüftools erstellt**

Datenformat DatML/RES

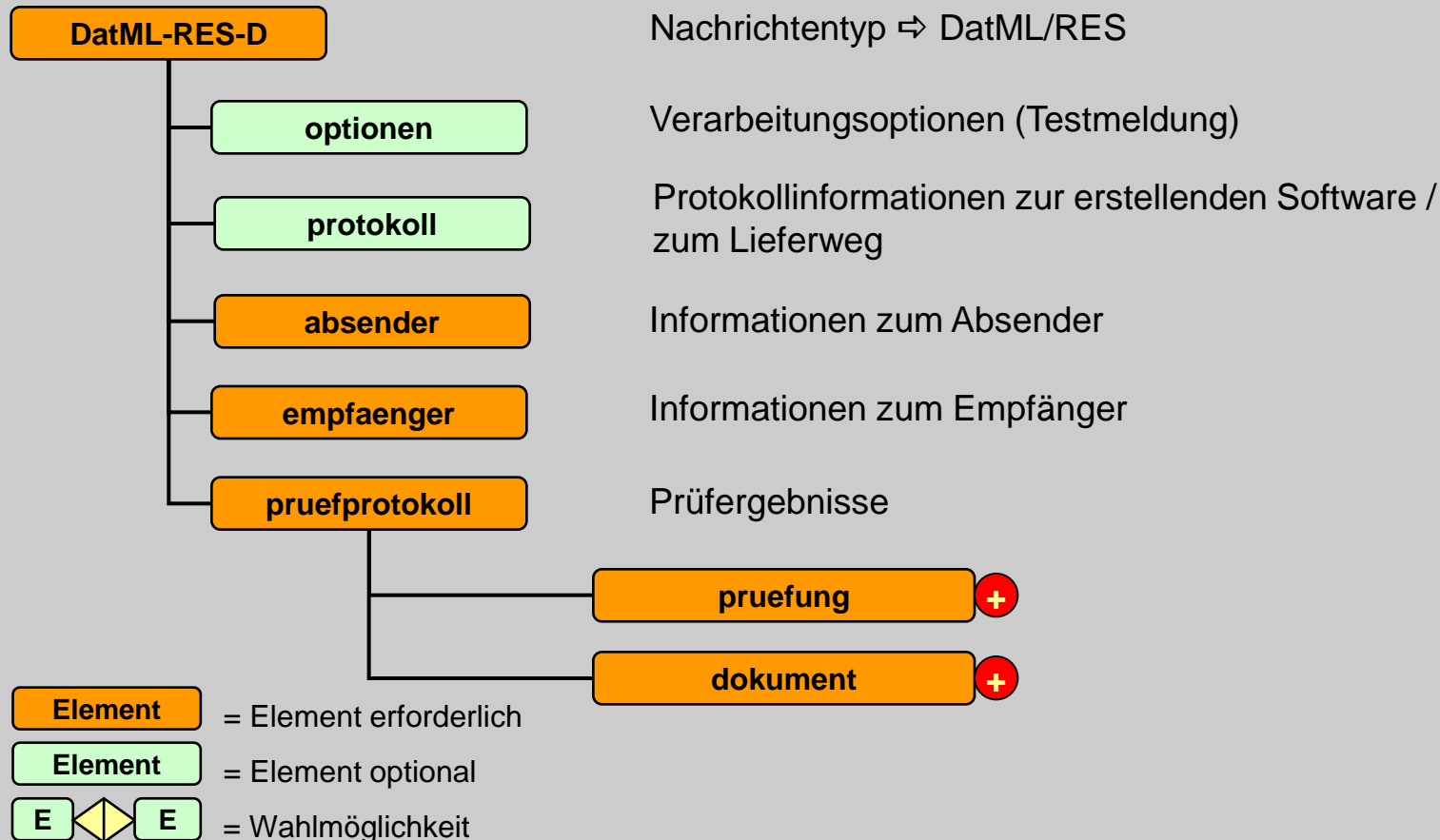
- **Das Prüfprotokoll im Format DatML/RES ist hierarchisch gegliedert**
 - Dokumentebene
 - Angaben zu
 - Verarbeitung (Kennzeichnung von Testmeldung)
 - Protokollierung (Kennzeichnung der erstellenden Software)
 - Absender (entspricht dem Empfänger der geprüften Gewerbemeldung)
 - Empfänger (entspricht dem Absender der geprüften Gewerbemeldung)
 - Prüfprotokollebene
 - Angaben zur Prüfung
 - Prüfstufe (verwendete Prüfstufe (1 - 4))
 - Dokumenttyp (Format der geprüften Datenlieferung)
 - Syntax (Prüfstatus der Syntaxprüfung - [Prüfstufe 1](#))
 - Semantik (Prüfstatus der Semantikprüfung (Validität) - [Prüfstufe 2](#))
 - Autorisierung (Prüfung der Angaben zur Identifikation und Autorisierung der Verfahrensteilnehmer, z.B. Kennungen und Adressen - [Prüfstufe 3](#))
 - Daten (Prüfung der erhebungsspezifischen Daten der Gewerbeanzeigen - [Prüfstufe 4](#))

Datenformat DatML/RES

- **Das Prüfprotokoll im Format DatML/RES ist hierarchisch gegliedert**
 - Prüfprotokollebene
 - Angaben zum Dokument
 - Verarbeitungshinweis („Hinweis: Ihre Datenlieferung wurde angenommen.“)
 - Erläuterungstext („Die Datenlieferung wird verarbeitet.“)
 - Dateneingang (enthält Informationen zum Zeitpunkt des Dateneingangs, zum Eingangsstempel, zur Anwendung ...)
 - Fehler (Fehlerliste der Dokumentenebene)
 - Nachricht
 - Fehler (Fehlerliste der Nachrichtenebene)
 - Meldung
 - Verarbeitungshinweis („Hinweis: Ihre Datenlieferung wurde angenommen.“)
 - Erläuterungstext („Die Datenlieferung wird verarbeitet.“)
 - Prüfung (Angaben zur Prüfung - Prüfstufe, sowie Angaben zum Prüfstatus zu Syntax, Semantik, Autorisierung und Daten auf Meldungs- bzw. Datenebene)
 - Erhebung (xGewerbeanzeige '0004')
 - Berichtszeitraum
 - Fehler (Fehlerliste der Meldungs- bzw. Datenebene)

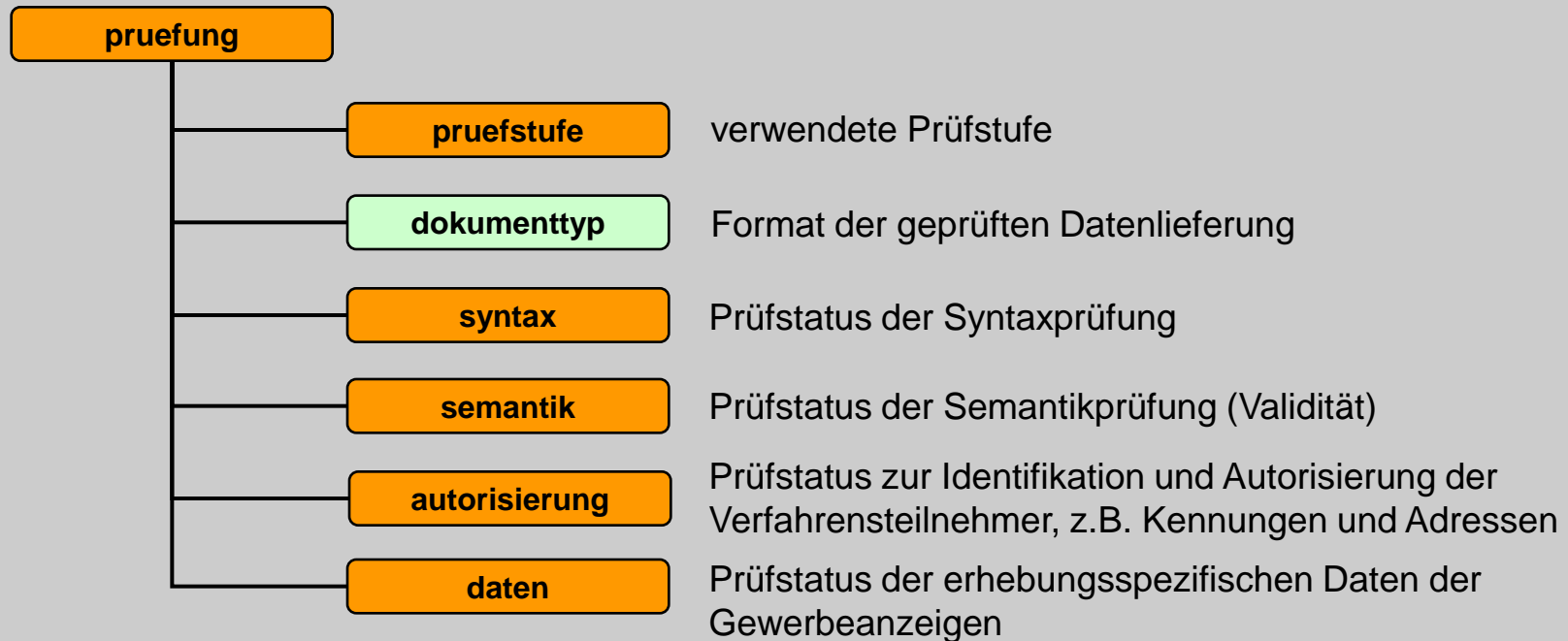
Datenformat DatML/RES


■ Aufbau des DatML/RES-Datenformates



Datenformat DatML/RES

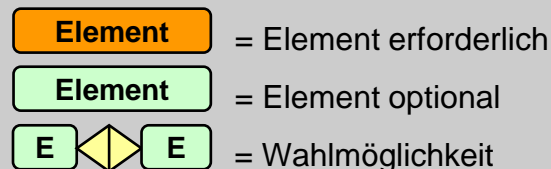
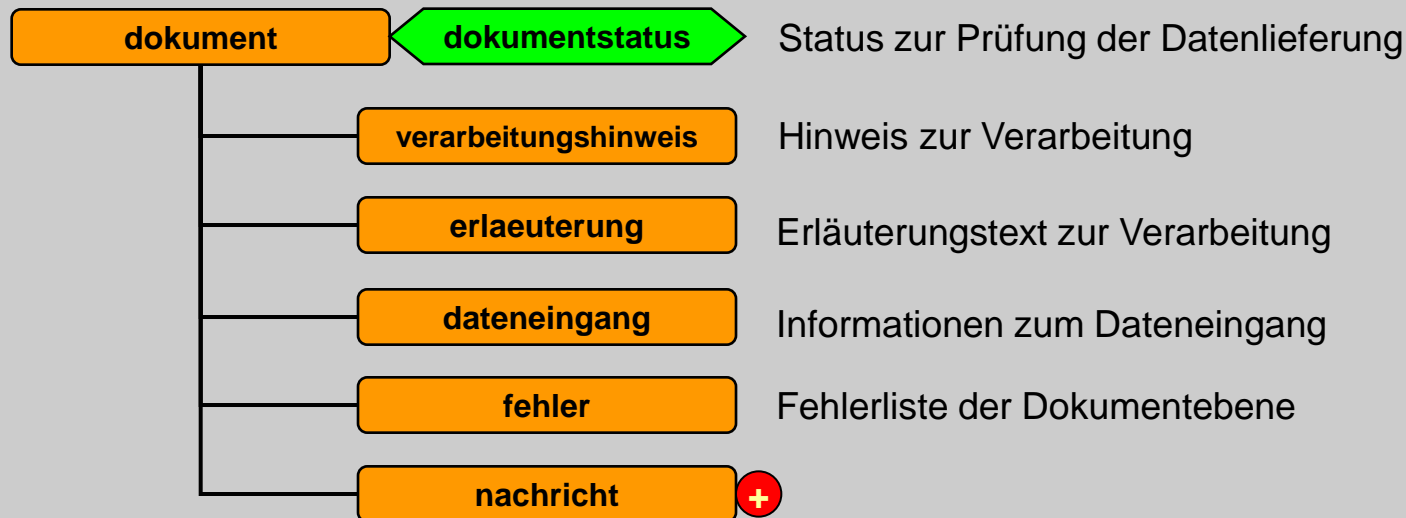
▪ Aufbau <pruefung>



- Element** = Element erforderlich
- Element** = Element optional
- E**  **E** = Wahlmöglichkeit

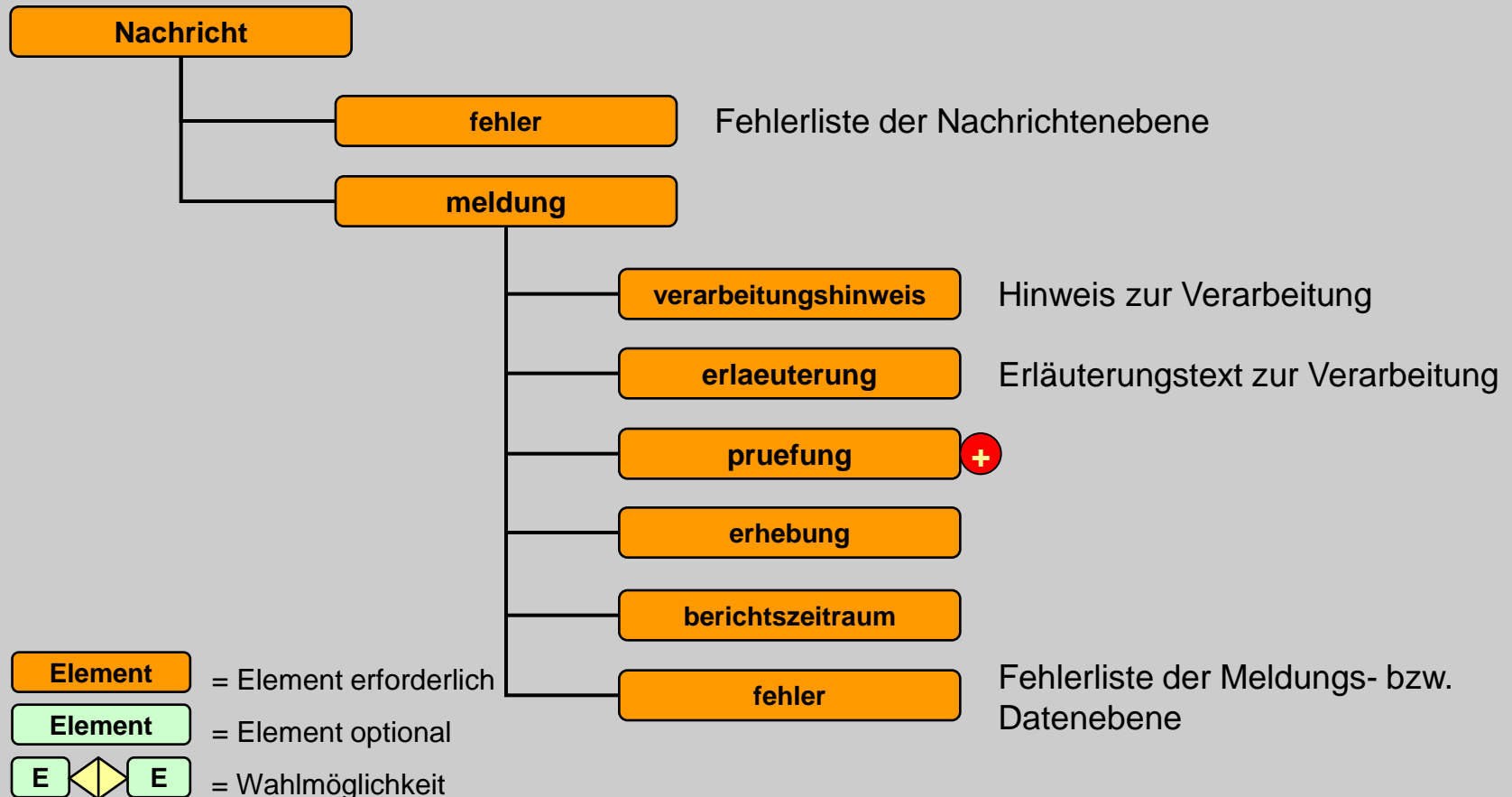
Datenformat DatML/RES

▪ Aufbau *<dokument>*



Datenformat DatML/RES

▪ Aufbau *<nachricht>*



Datenformat DatML/SDF

Datenformat DatML/SDF

- **DatML/SDF ist ein XML-Format zur Beschreibung von Erhebungen im DatML/RAW-Format**
 - dient als Datenmodell für die Erhebung
- **Definiert Erhebung mit zu liefernden *Hilfsmerkmalen, Merkmalen* und *Merkmalsgruppen***
- **Beschreibt Regeln zur Belegung von *Hilfsmerkmalen, Merkmalen* und *Merkmalsgruppen***
- **Beschreibung von Gültigkeitsprüfungen zu *Hilfsmerkmalen, Merkmalen* und *Merkmalsgruppen***
- **Einsatzspektrum**
 - Steuerung generischer Anwendungen, die DatML/RAW erzeugen
 - Steuerung generischer Anwendungen, die DatML/RAW prüfen / verarbeiten
 - Unterstützt die Entwicklung generischer Software

Datenformat DatML/SDF

■ **Aufbau des Datenformates**

- Beschreibung allgemeiner Dokumenteigenschaften
 - Kurzbeschreibung des Dokumentinhaltes
 - Copyright
 - Adress- und Kontaktdaten des Autors
 - Datum und Uhrzeit der Dokumenterzeugung
- Definition des statistischen Kontextes
 - Objektbeschreibungen - Beschreibung des aktuellen Objektes (Selbstreferenz)
 - ObjektID („Erhebung:109951201329692“)
 - Objektname („Gewerbeanzeigenstatistik_ab_2016“)
 - Klasse oder Typ des Objektes („de.statspez.material.Erhebung“)
 - Version des Objektes („1“)

Datenformat DatML/SDF

- **Aufbau des Datenformates**

- Definition des statistischen Kontextes

- Erhebungseigenschaften - Administrative Beschreibung einer Erhebung
 - Statistikidentifikator („0004“)
 - Bezeichnung der Statistik („Gewerbeanzeigenstatistik_ab_2016“)
 - Version der Erhebung („1“)
 - Identifikator des Erhebungsverfahrens (Ressourcenschlüssel „00041640019999000001“)
 - Berichtszeitraum
 - Bezeichnung des Berichtsgebiets (Name: „Bundesgebiet“)
 - Bezeichnung der Erhebungsobjekte
 - Periodizität (hier: Monatlich)

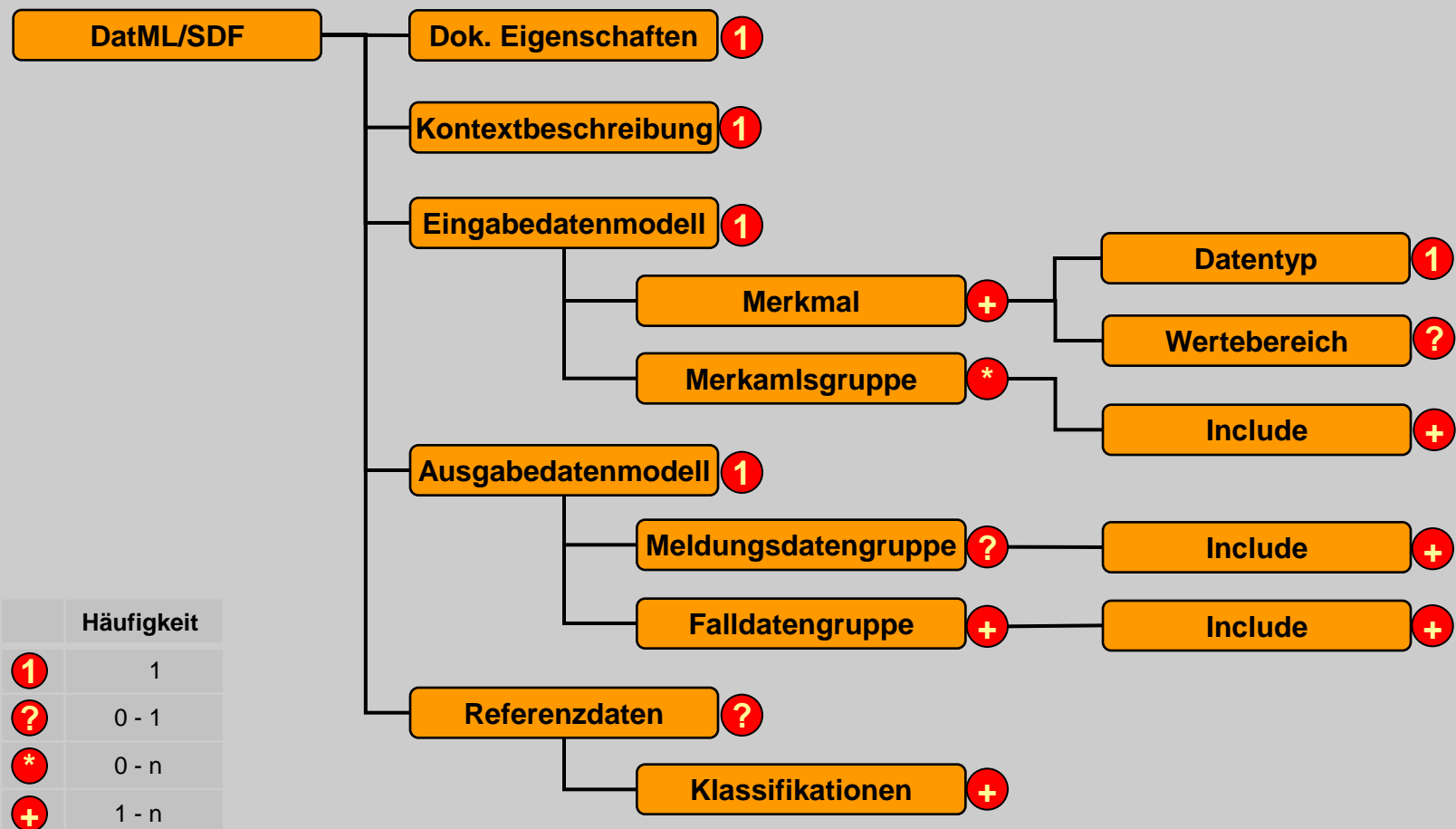
Datenformat DatML/SDF

■ **Aufbau des Datenformates**

- Definition des Eingabedatenmodells
 - Definition der Merkmale
 - Definition von Merkmalsgruppen mit den Ausgabegruppen
- Definition des Ausgabedatenmodells
 - Definition der Struktur der zu erzeugenden Meldung
 - Definition der Meldungsdatengruppe
 - meldungsglobale Daten ⇨ entsprechen den Hilfsmerkmalen in DatML/RAW
 - Definition von Vorbedingung für Merkmale
 - Inkludierung der Merkmale z.B. „BerichtseinheitID“
 - Definition einer Falldatengruppe
 - Daten der „Berichtsfälle“ ⇨ Daten der Dokumentenebene von DatML/RAW
 - Definition eines Identifikators zur Auswahl der Fallgruppe
 - Definition von Vorbedingungen für Merkmale
 - Definition von Abhängigkeiten
 - Inkludierung der Merkmalsgruppen

Datenformat DatML/SDF

■ Aufbau des DatML/SDF-Datenformates



Datenformat DatML/SDFmeta

Datenformat DatML/SDFmeta

- **DatML/SDFmeta ist ein XML-Format zur Beschreibung der Metadaten im DatML/RAW-Format**
 - erweitert DatML/SDF um Datenmodell der zu liefernden Metadaten (nicht statistikspezifischen Daten einer Meldung)
- **generische Erzeugung unterschiedlicher DaML/RAW-Versionen mittels unterschiedlicher DatML/SDFmeta-Dateien**
 - DatML/SDFmeta 2.0 ⇒ DatML/RAW Version 2.0
 - DatML/SDFmeta 2.1 ⇒ DatML/RAW Version 2.1
- **DatML/SDFmeta dient der generischen Erzeugung statistischer Meldungen**
 - Ausgabe der generischen Erzeugung wird durch Anweisungen in DatML/SDFmeta gesteuert
- **Einsatzspektrum**
 - Steuerung generischer Anwendungen, die DatML/RAW erzeugen
 - Unterstützt die Entwicklung generischer Software

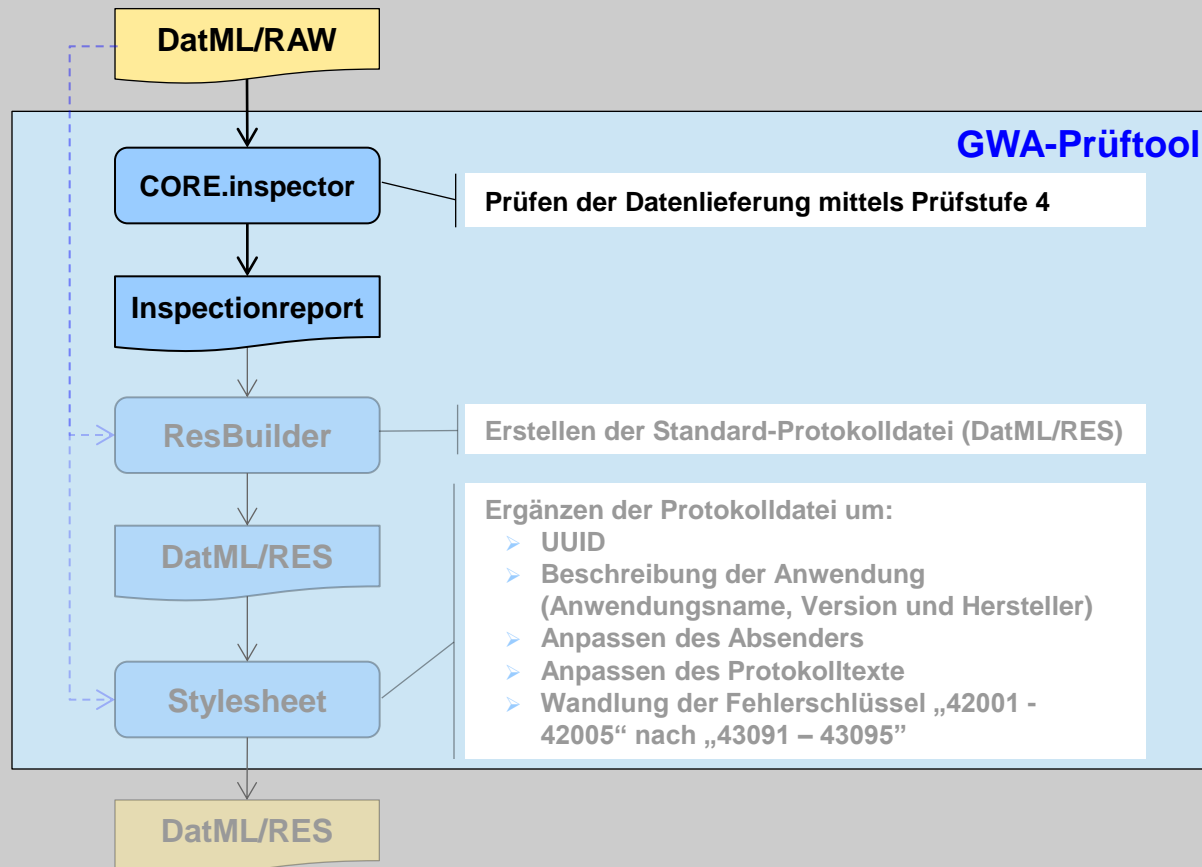
1. Datenformate DatML/RAW, DatML/RES, DatML/SDF und DatML/SDFmeta

2. Funktionsweise und Schnittstellen des GWA-Prüftools

3. Generisch unterstützte Erzeugung von Gewerbemeldungen

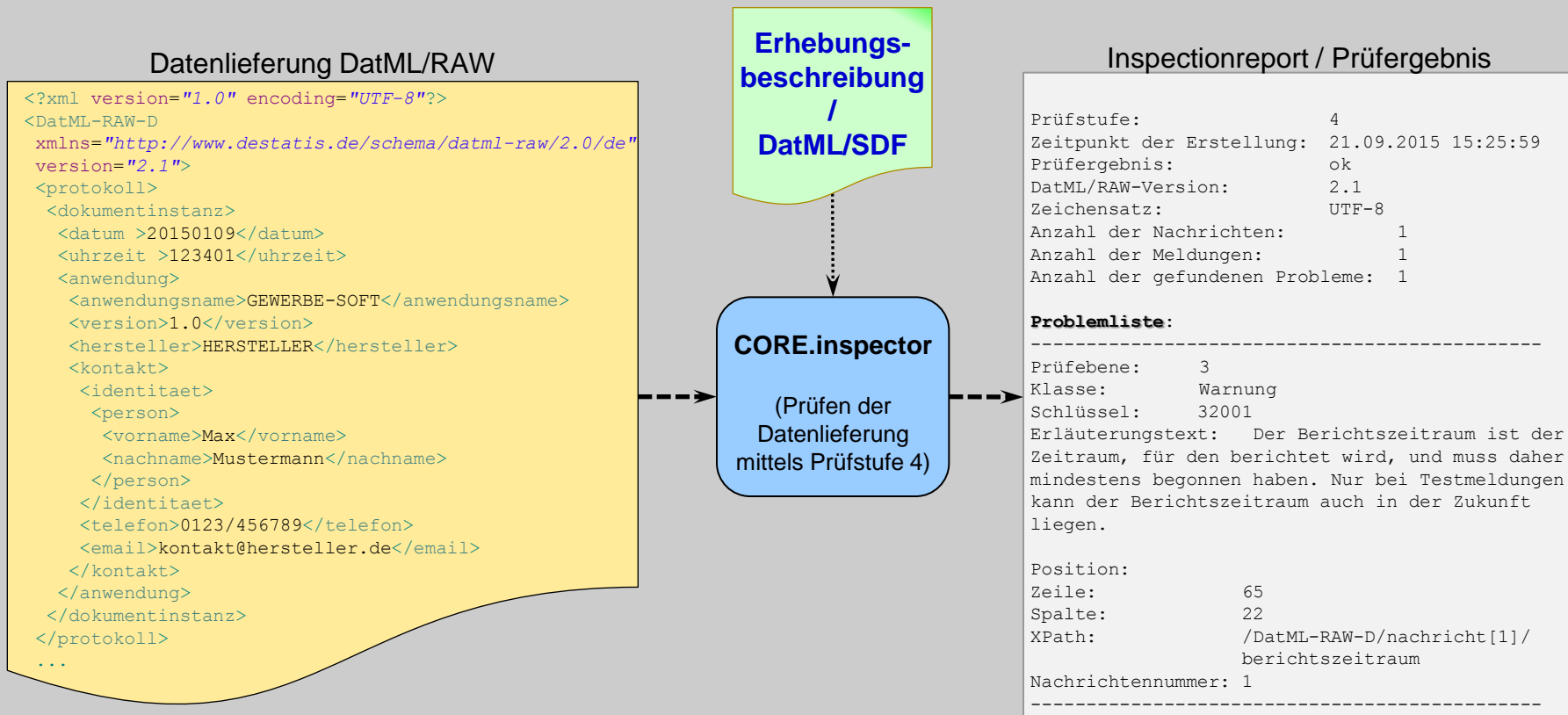
Funktionsweise des GWA-Prüftool

- **Darstellung des Prüfablaufs bei Einsatz des GWA-Prüftools**
 - Prüfung der Datenlieferung



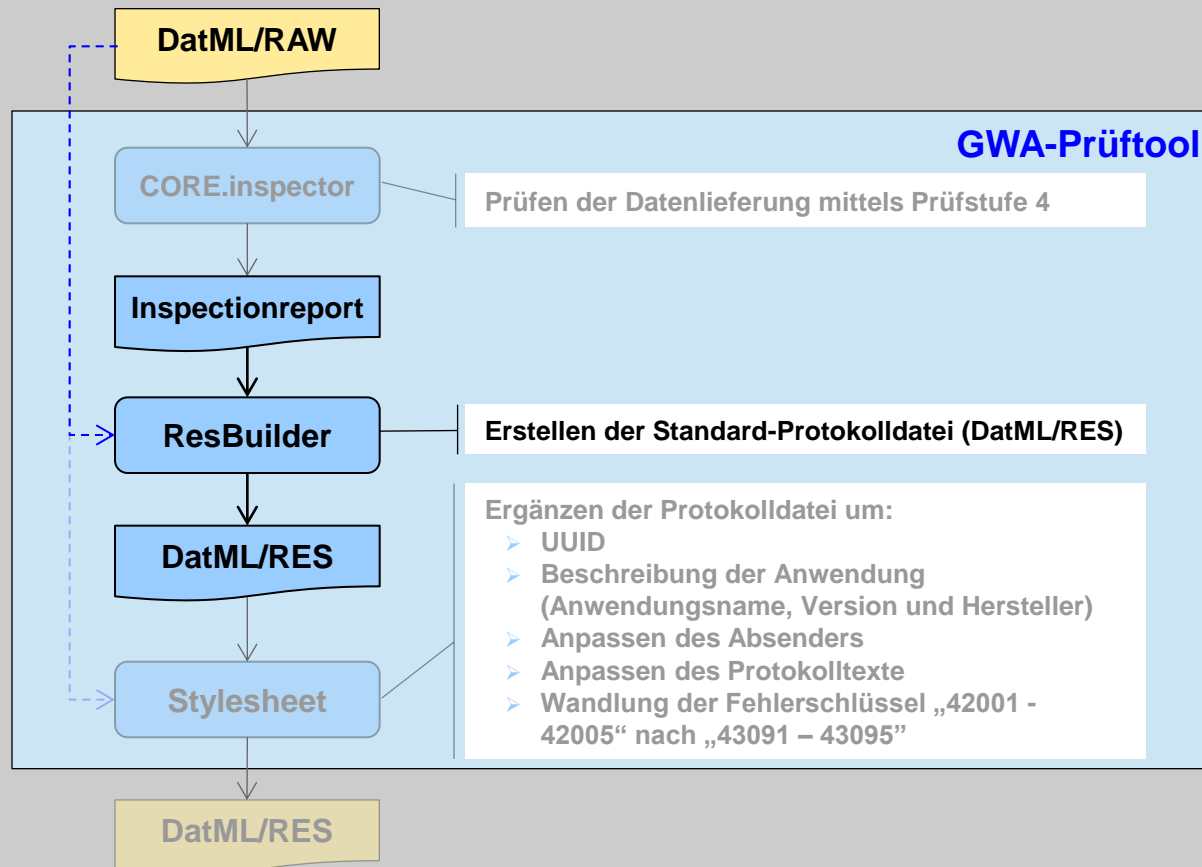
Funktionsweise des GWA-Prüftool

- **Darstellung des Prüfablaufs bei Einsatz des GWA-Prüftools**
 - Prüfung der Datenlieferung



Funktionsweise des GWA-Prüftool

- **Darstellung des Prüfablaufs bei Einsatz des GWA-Prüftools**
 - Erstellung Standard-Protokolldatei



Funktionsweise des GWA-Prüftool

- **Darstellung des Prüfablaufs bei Einsatz des GWA-Prüftools**
 - Erstellung Standard-Protokolldatei

Inspectionreport

```

Prüfstufe: 4
Zeitpunkt der Erstellung: 21.09.2015 15:25:59
Prüfergebnis: ok
DatML/RAW-Version: 2.1
Zeichensatz: UTF-8
Anzahl der Nachrichten: 1
Anzahl der Meldungen: 1
Anzahl der gefundenen Probleme: 1

Problemliste:
-----
Prüfebene: 3
Klasse: Warnung
Schlüssel: 32001
Erläuterungstext: Der Berichtszeitraum ist der
Zeitraum, für den berichtet wird, und muss daher
mindestens begonnen haben. Nur bei Testmeldungen
kann der Berichtszeitraum auch in der Zukunft
liegen.

Position:
Zelle: 65
Spalte: 22
XPath: /DatML-RAW-D/nachricht[1]/
berichtszeitraum
Nachrichtennummer: 1
    
```

Datenlieferung

```

<?xml version="1.0" encoding="UTF-8"?>
<DatML-RAW-D
  xmlns="http://www.destatis.de/schema/datml-raw/2.0/de"
  version="2.1">
  <protokoll>
    <dokumentinstanz>
      <datum>20150109</datum>
      <uhrzeit>123401</uhrzeit>
      <anwendung>
        <anwendungsname>GEWERBE-SOFT</anwendungsname>
        <version>1.0</version>
        <hersteller>HERSTELLER</hersteller>
        <kontakt>
          <identitaet>
            <person>
              <vorname>Max</vorname>
              <nachname>Mustermann</nachname>
            </person>
          </identitaet>
          <telefon>0123/456789</telefon>
          <email>kontakt@hersteller.de</email>
        </kontakt>
      </anwendung>
    </dokumentinstanz>
  </protokoll>
    
```

ResBuilder

(Erstellen der
Standard-
Protokolldatei
(DatML/RES))

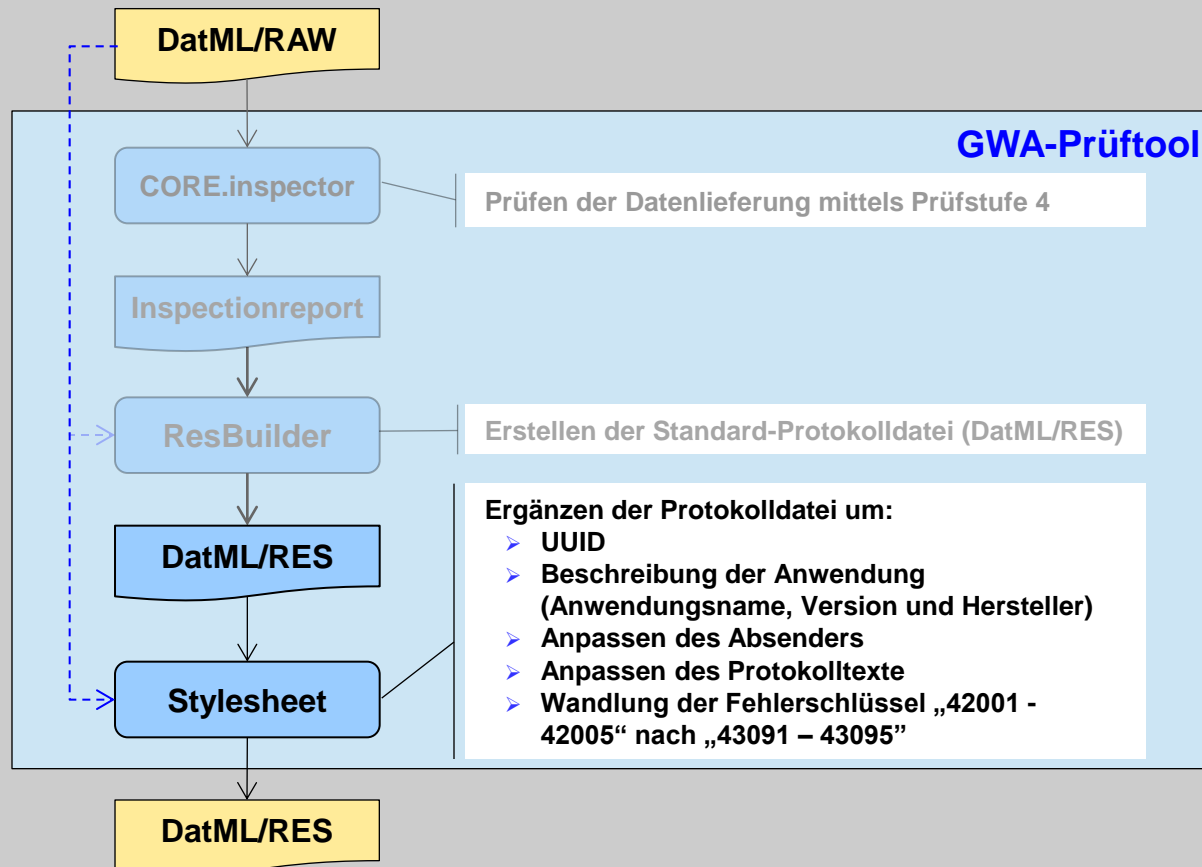
Standard-Protokolldatei (DatML/RES)

```

<?xml version="1.0" encoding="UTF-8"?>
<DatML-RES-D
  xmlns="http://www.destatis.de/schema/datml-raw/2.0/de"
  version="2.1">
  <protokoll>
    <dokumentinstanz>
      <datum>20150312</datum>
      <uhrzeit>091529</uhrzeit>
      <anwendung>
        <anwendungsname>ResBuilder</anwendungsname>
        <version>1.1.27</version>
        <hersteller>
          Statistisches Bundesamt, C 304, 30.01.2014, 10:50
        </hersteller>
      </anwendung>
    </dokumentinstanz>
  </protokoll>
  <absender>
    <kennung klasse="DGUID">99</kennung>
    <externeIdentifikation klasse="xga">
      490700000000
    </externeIdentifikation>
  </absender>
  <empfaenger>
    <kennung klasse="MELDID">012345678901</kennung>
    <identifikation>
      <identitaet>
        ...
      </identitaet>
    </identifikation>
  </empfaenger>
    
```

Funktionsweise des GWA-Prüftool

- **Darstellung des Prüfablaufs bei Einsatz des GWA-Prüftools**
 - Ergänzung der Protokolldatei



Funktionsweise des GWA-Prüftool

- Darstellung des Prüfablaufs bei Einsatz des GWA-Prüftools
 - Ergänzung der Protokolldatei

Standard-Protokolldatei

```
<?xml version="1.0" encoding="UTF-8"?>
<DatML-RES-D
  xmlns="http://www.destatis.de/schema/datml-raw/2.0/de"
  version="2.1">
  <protokoll>
    <dokumentinstanz>
      <datum>20150312</datum>
      <uhrzeit>091529</uhrzeit>
      <anwendung>
        <anwendungsname>ResBuilder</anwendungsname>
        <version>1.1.27</version>
        <hersteller>
          Statistisches Bundesamt, C 304, 30.01.2014, 10:50
        </hersteller>
      </anwendung>
    </dokumentinstanz>
    </protokoll>
    <absender>
      <denung_klasse>"DGUID"-99</denung_klasse>
      <externelldentifikation_klasse>"xga">
        490700000000
      </externelldentifikation_klasse>
    </absender>
    <empfänger>
      <denung_klasse>"MELDID"-012345678901</denung_klasse>
      <identifikation>
        <identitaet>
          ...
        </identifikation>
      </empfänger>
    </absender>
  </DatML-RES-D>
```

Datenlieferung

```
<?xml version="1.0" encoding="UTF-8"?>
<DatML-RAW-D
  xmlns="http://www.destatis.de/schema/datml-raw/2.0/de"
  version="2.1">
  <protokoll>
    <dokumentinstanz>
      <datum>20150109</datum>
      <uhrzeit>123401</uhrzeit>
      <anwendung>
        <anwendungsname>GEWERBE-SOFT</anwendungsname>
        <version>1.0</version>
        <hersteller>HERSTELLER</hersteller>
        <kontakt>
          <identitaet>
            <person>
              <vorname>Max</vorname>
              <nachname>Mustermann</nachname>
            </person>
            <telefon>0123456789</telefon>
            <email>kontakt@hersteller.de</email>
          </kontakt>
        </anwendung>
      </dokumentinstanz>
    </protokoll>
```

Stylesheet

Ergänzen der Protokolldatei um:

- UUID
- Beschreibung der Anwendung (Anwendungsname, Version und Hersteller)
- Anpassen des Absenders
- Anpassen des Protokolltextes
- Wandlung der Fehlerschlüssel „42001 - 42005“ nach „43091 – 43095“

Protokolldatei (DatML/RES)

```
...
<dokument dokumentstatus="abgewiesen"
  pruefstatus="fehlerhaft">
  <verarbeitungshinweis>
    Hinweis: Ihre Datenlieferung wurde abgewiesen.
  </verarbeitungshinweis>
  <erlaeuterung>
    Die Datenlieferung wird nicht verarbeitet.
  </erlaeuterung>
  <dateneingang>
    ...
    <anwendung>
      <anwendungsname>GEWERBE-SOFT</anwendungsname>
      <version>1.0</version>
      <hersteller>HERSTELLER</hersteller>
    </anwendung>
    ...
    <fehler_klasse="INSP">
      <schluessel>43010</schluessel>
      <gewicht>ERROR</gewicht>
      <text>
        Der Wert des Merkmals liegt nicht im definierten Wertebereich.
      </text>
      <position zeileSpalte="81/22" format="xpath">
        /DatML-RAW-D/nachricht[1]/datensegment/
        satz [mm[@name='UUID'] [wert=
          '589e8421-e30a-14d4-a716-646655440211']] /
        mm [@name="MeldungArt"] /wert
      </position>
      <merkmal>MeldungArt</merkmal>
    </fehler>
  </dateneingang>
  ...
```

Schnittstellen des GWA-Prüftool

- **Systemvoraussetzungen**

- Java-Laufzeitumgebung (JRE) ab Version 7

- **Erforderliche Verzeichnisse**

- Eingabeverzeichnis Default: „in“
- Ausgabeverzeichnis Default: „out“
- Verzeichnis der Log-Dateien Default: „logs“
- Erhebungsbeschreibungsverzeichnis Default: „sdf“
- XSLT-Verzeichnis zur Ablage des Transformationsscripts Default: „xslt“

- **Erforderliche Dateinamen**

- zu prüfende XSTATISTIK-Meldung
- zu erzeugende Protokolldatei Default: „DATEINAME-prot.xml“

Schnittstellen des GWA-Prüftool

- **Weitere erforderliche Angaben**

- Eingangsstempel zur eindeutigen Kennzeichnung des Dateneingangs
Default: Dateiname der zu prüfenden XSTATISTIK-Meldung
- Prüfstufe des CORE.inspectors
Default: formale Datenprüfung

Schnittstellen des GWA-Prüftool

- **Mögliche Verwendung von Konfigurationsparameter für**
 - Verzeichnisse
 - Eingabeverzeichnis
 - Ausgabeverzeichnis
 - Verzeichnis der Log-Dateien
 - Erhebungsbeschreibungsverzeichnis
 - XSLT-Verzeichnis zur Ablage des Transformationsscripts
 - Dateinamen
 - der zu prüfenden XSTATISTIK-Meldung
 - der zu erzeugenden Protokolldatei
 - Eingangsstempel zur eindeutigen Kennzeichnung des Dateneingangs
 - Prüfstufe des CORE.inspectors

Schnittstellen des GWA-Prüftool

- **Mögliche Verwendung einer Properties-Datei zur Konfiguration**
 - des Eingabeverzeichnisses
 - des Ausgabeverzeichnis
 - des Verzeichnisses der Log-Dateien
 - des Erhebungsbeschreibungsverzeichnisses
 - des XSLT-Verzeichnisses zur Ablage des Transformationsscripts

Schnittstellen des GWA-Prüftool

■ Erforderliche Anpassungen

■ Beschreibung der Anwendung zur Ergänzung der Protokolldatei anpassen

■ in Konfigurationsdatei ‚res2res-gewerbe.config.xml‘

- Anwendungsname
- Versionsbezeichnung der Anwendung
- Hersteller der Anwendung

■ Beispiel

```
...  
<anwendung>  
  <name>ANWENDUNGSNAME</name>  
  <version>VERSION_DER_ANWENDUNG</version>  
  <hersteller>HERSTELLER_DER_ANWENDUNG</hersteller>  
</anwendung>  
...
```

Schnittstellen des GWA-Prüftool

- **Beispiel eines Aufrufs zur Prüfung der XStatistik-Meldung
,BeispielSpezifikation.xml‘,**
 - Verwendung der Default-Einstellungen

```
GWAPrueftool tool = new GWAPrueftool();  
tool.generateProtokoll(„BeispielSpezifikation.xml“);
```
 - Verwendung der Default-Einstellungen unter Modifikation des Eingabeverzeichnis und Übergabe des Eingangsstempels

```
GWAPrueftool tool = new GWAPrueftool();  
generateProtokoll(„inputDir“, „BeispielSpez.xml“, „eingangsstempel“)
```
 - Verwendung von Properties aus Properties-Datei

```
GWAPrueftool tool = new GWAPrueftool();  
tool.generateProtokollWithProperties(„BeispielSpez.xml“, „PROPS.DATEI“);
```

Schnittstellen des GWA-Prüftool

■ Setter-Methoden

- Setzt den InspectionLevel
`setInspectionLevel(int inspectionLevel);`
- Setzt das Logs-Verzeichnis
`setLogsDir(java.lang.String logsDir);`
- Setzt das Ausgabeverzeichnis zur Ablage der erzeugten Protokolldateien
`setOutputDir(java.lang.String outputDir);`
- Setzt den Dateinamen der Protokolldatei
`setOutputResFile(java.lang.String outputResFile);`
- Setzt das Erhebungsbeschreibungsverzeichnis
`setSurveyDir(java.lang.String surveyDir);`
- Setzt das Verzeichnis der Transformationsdatei (res2res-gewerbe.xsl) und der dazugehörigen Konfigurationsdatei (res2res-gewerbe.config.xml)
`setXsltDir(java.lang.String xsltDir);`

Schnittstellen des GWA-Prüftool

■ Getter-Methoden

- Liefert den aktuell eingestellten InspectionLevel
`getInspectionLevel();`
- Liefert das aktuell eingestellte Logs-Verzeichnis
`getLogsDir();`
- Liefert das aktuell eingestellte Ausgabeverzeichnis der Protokolldateien
`getOutputDir();`
- Liefert den aktuell eingestellten Dateinamen der Protokolldatei
`getOutputResFile();`
- Liefert das aktuell eingestellte Erhebungsbeschreibungsverzeichnis
`getSurveyDir();`
- Liefert das aktuell eingestellte Verzeichnis der Transformationsdatei (res2res-gewerbe.xsl) und der Konfigurationsdatei (res2res-gewerbe.config.xml)
`getXsltDir();`

Schnittstellen des GWA-Prüftool

- **Getter-Methoden**

- Liefert die Version des GWA-Prüftools

`getVersion();`

- Liefert den Erstellungszeitpunkt des GWA-Prüftools

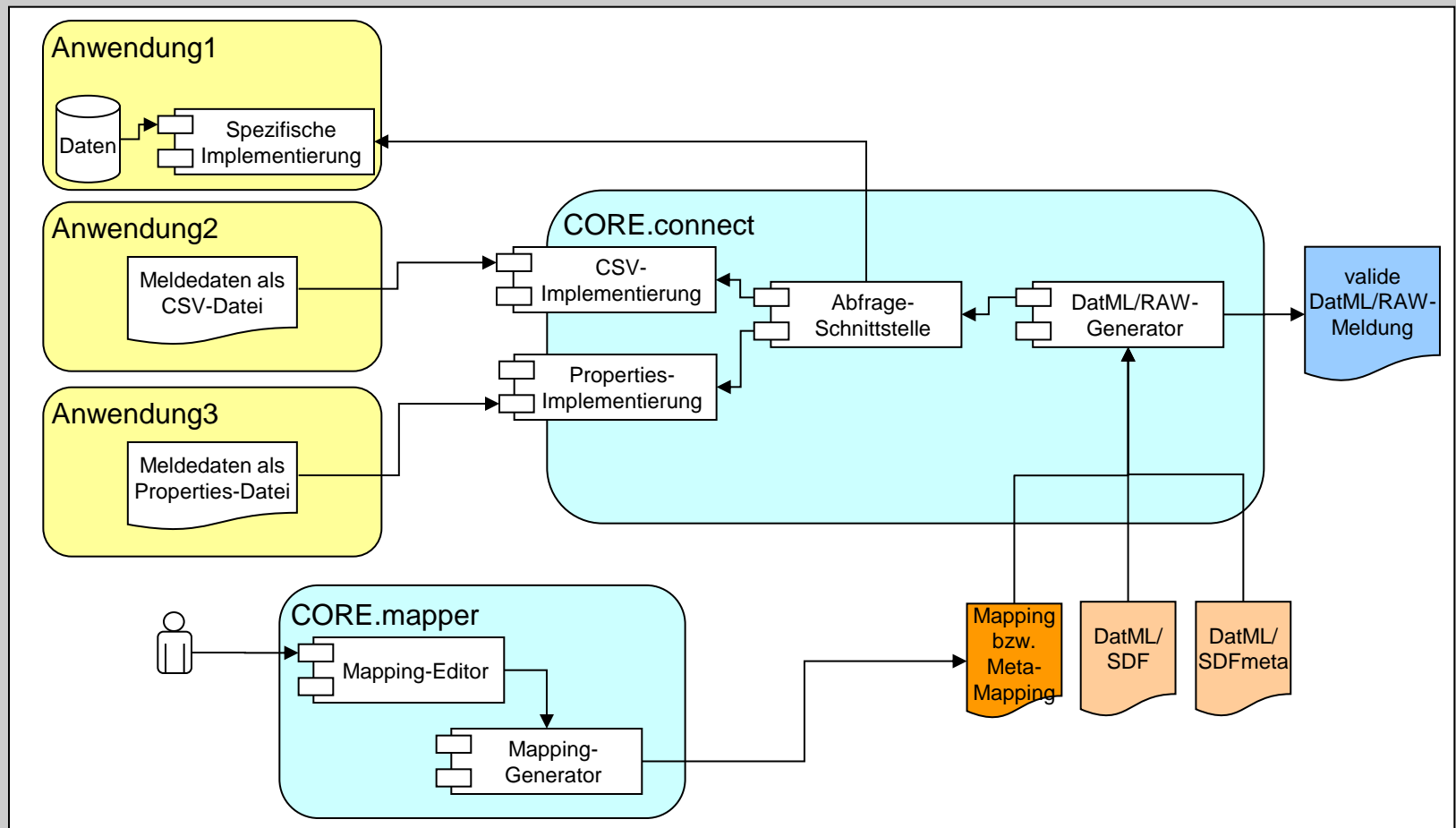
`getVersionInfo();`

1. Datenformate DatML/RAW, DatML/RES, DatML/SDF und DatML/SDFmeta
2. Funktionsweise und Schnittstellen des GWA-Prüftools

3. Generisch unterstützte Erzeugung von Gewerbemeldungen

Generisch unterstützte Erzeugung

- Spezifische Implementierung mittels Softwarebibliothek CORE.connect



Generisch unterstützte Erzeugung

■ Metadaten-Mapping-Datei

- Metadaten-Mapping zu Metadaten des Lieferdokumentes
- Zuordnung von Meta-Merkmalen zu benutzerspezifischen Feldinformationen
- Generierungsprozess zusätzlich steuern
 - Konfigurationsmöglichkeit der Implementierung der Abfrage-Schnittstelle von außen

Name	Benutzerspez. Name	Position	Anzahl	Standardwert
Metadaten				
Dokumentebene				
Lieferungsart		1		
Anwendung			1	
Anwendungsname		2		
Version		3		
Hersteller		4		
Kontakt			1	
Absender			1	
Kennung		28		
Organisation			1	
Person			1	
Adresse			1	
ExterneIdentifikation			1	
Kontakt			1	
Korrektur			1	
Telefon		115		
Email		116		
Fax		117		
URL		118		
Nachrichtenebene				
ErhebungsID		1		
Berichtszeitraum		2		
Berichtsempfänger		3		
Berichtspflichtiger			1	
Material			1	
Dezimalzeichen		99		
TausenderTrennzeichen		100		
MeldungsID		101		
Meldungsart		102		
OrigMeldungsID		103		

Generisch unterstützte Erzeugung

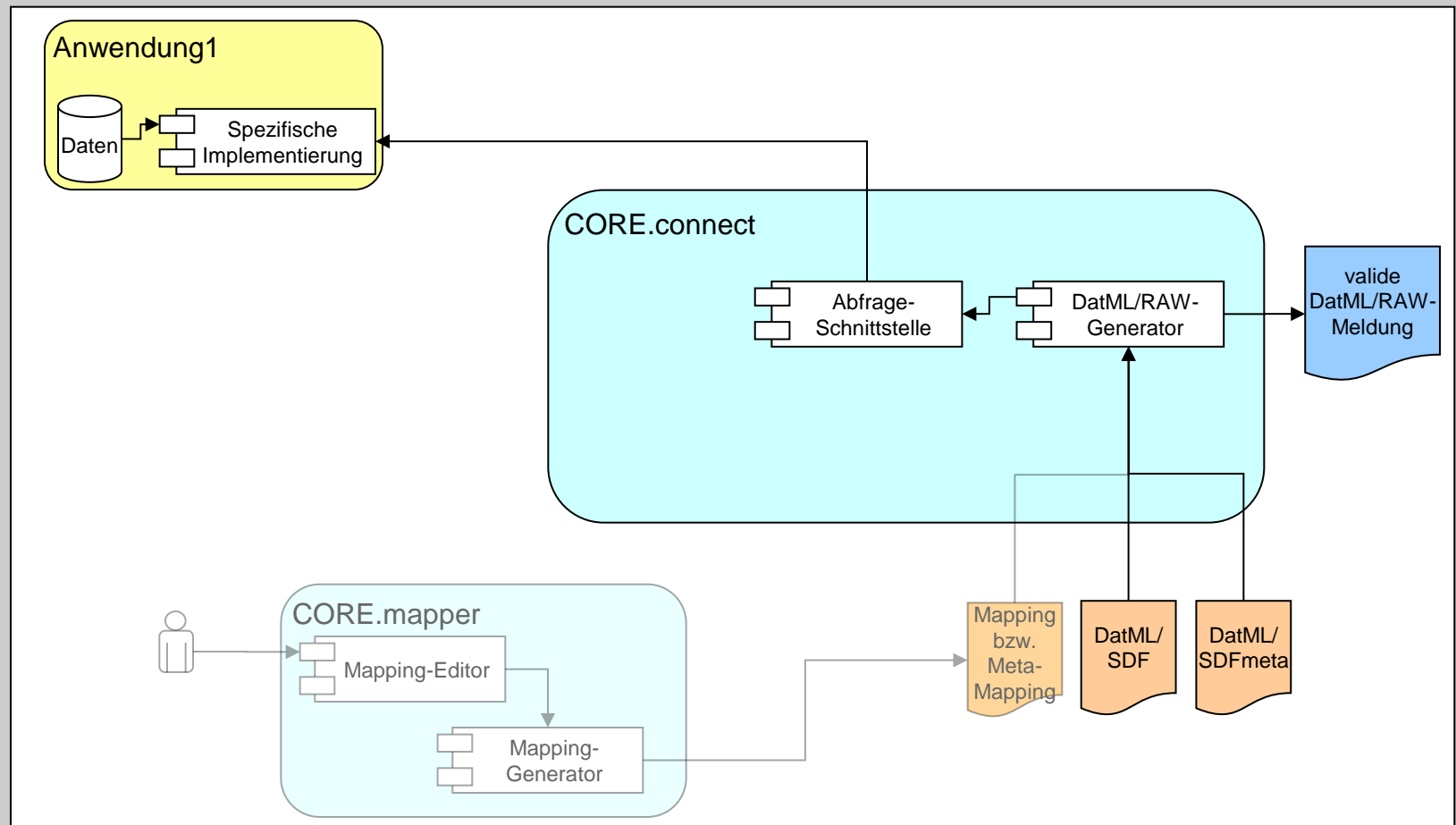
■ Mapping-Datei

- Mapping einer konkreten Erhebung
- Zuordnung von Hilfsmerkmalen und Merkmalen zu benutzerspezifischen Feldinformationen
- Generierungsprozess zusätzlich steuern
 - Konfigurationsmöglichkeit der Implementierung der Abfrage-Schnittstelle von außen

Name	Benutzerspez. Name	Position	Anzahl	Standardwert
Gewerbesteuer (komplett)				
BerichtseinheitID		1		
BerichtseinheitIDZusatz		2		
GemeindeName		3		
Bemerkungen		4		
BerEmpfaenger		5		
Satzart		1		
IHKID				
HWKID				
STAD				
ISID				
DGUID				
EAID				
FAID				
RGID				
ZVID				
ASID				
ALGID				
LUEID				

Generisch unterstützte Erzeugung

- Spezifische Implementierung mittels Softwarebibliothek CORE.connect

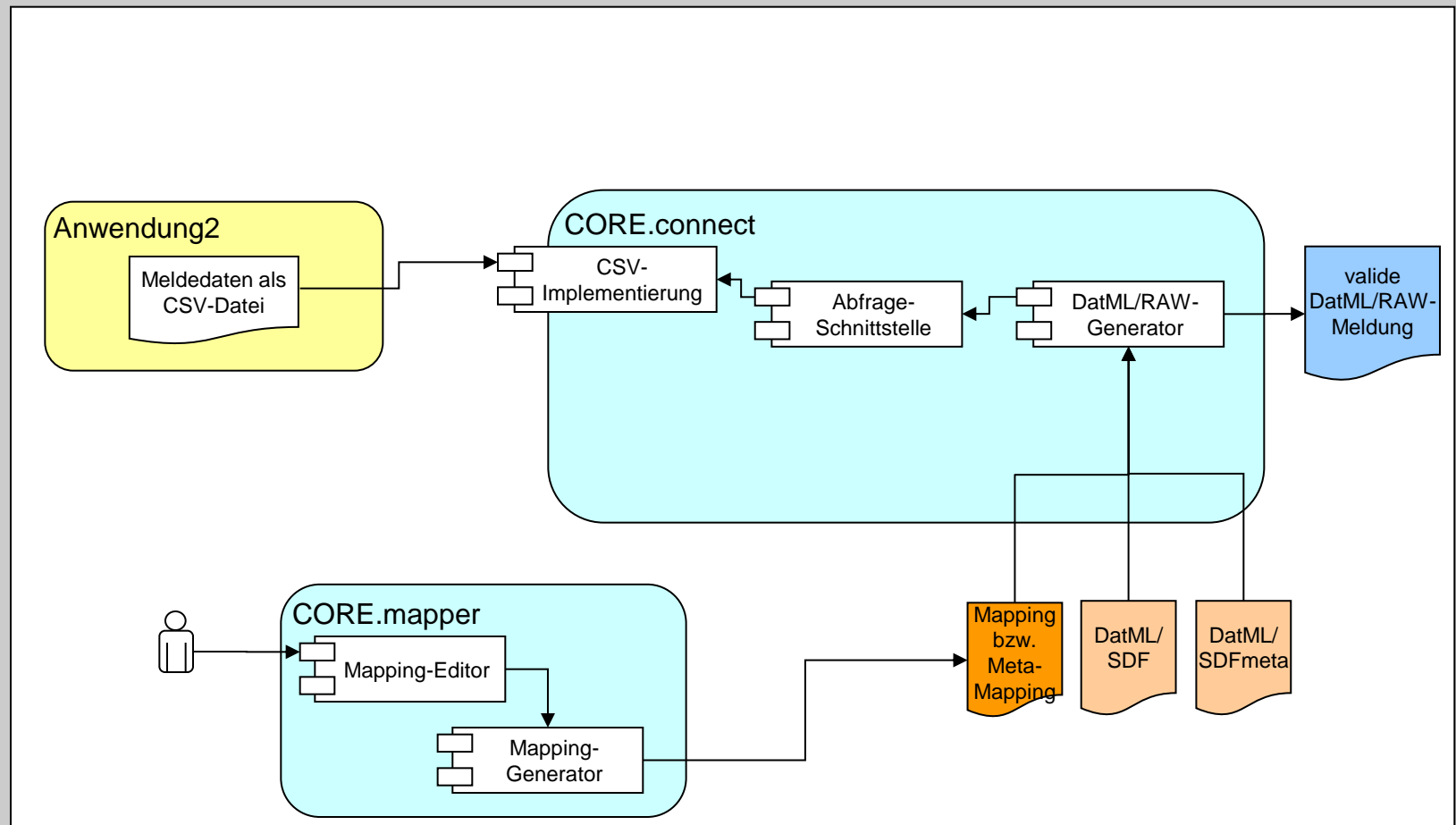


Generisch unterstützte Erzeugung

- **Spezifische Implementierung mittels Softwarebibliothek CORE.connect**
 - Interface DataProvider in anwenderspezifischer Software einbinden (Schnittstelle für den Zugriff auf die Metadaten/Daten)
 - Methoden des Interfaces DataProvider implementieren
 - getDataProvider (StructureDescriptor structure)
Abfrage der Substrukturen (Nachrichten, Sätze, Merkmalsgruppen oder MetaMerkmalsgruppen)
 - getFieldValue (FieldDescriptor field)
Abfrage der Metadaten und der relevanten Daten
 - close()
Gibt die von diesem DataProvider erzeugten Ressourcen frei (Schliessen von Streams etc.)

Generisch unterstützte Erzeugung

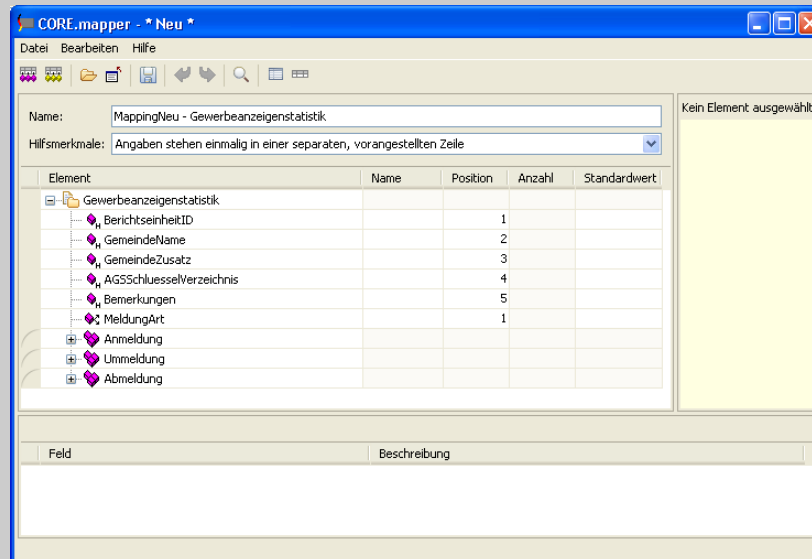
- **Meldedaten als CSV-Datei**



Generisch unterstützte Erzeugung

■ Meldedaten als CSV-Datei

- Mittels CORE.mapper → Metadaten-Mapping und Mapping erzeugen
 - DatML/SDFmeta bzw. DatML/SDF in CORE.mapper einlesen
 - Datei → Neu → Metadaten-Mapping (für DatML/SDFmeta)
 - Datei → Neu → Mapping (für DatML/SDF)



- Metadaten-Mapping für SDFmeta erzeugen mittels Speichern
- Mapping für SDF erzeugen mittels Speichern

Generisch unterstützte Erzeugung

- **Meldedaten als CSV-Datei**

- Erforderlicher struktureller Aufbau einer CSV-Datei

- Erste Zeile enthält die Metadaten des Dokuments

```
2;Anwendung;10;Firma;Kennung;Organisation;Str.;3;12345;Stadt;Mustermann;Paul;12345;test@test.de
```

- Zweite Zeile enthält die Metadaten der Nachricht

```
0004;200701;00;1;2;1234;...
```

- Dritten Zeile folgen (je nach Konfiguration des übergebenen Mappings) ggf. die Hilfsmerkmale der Nachricht

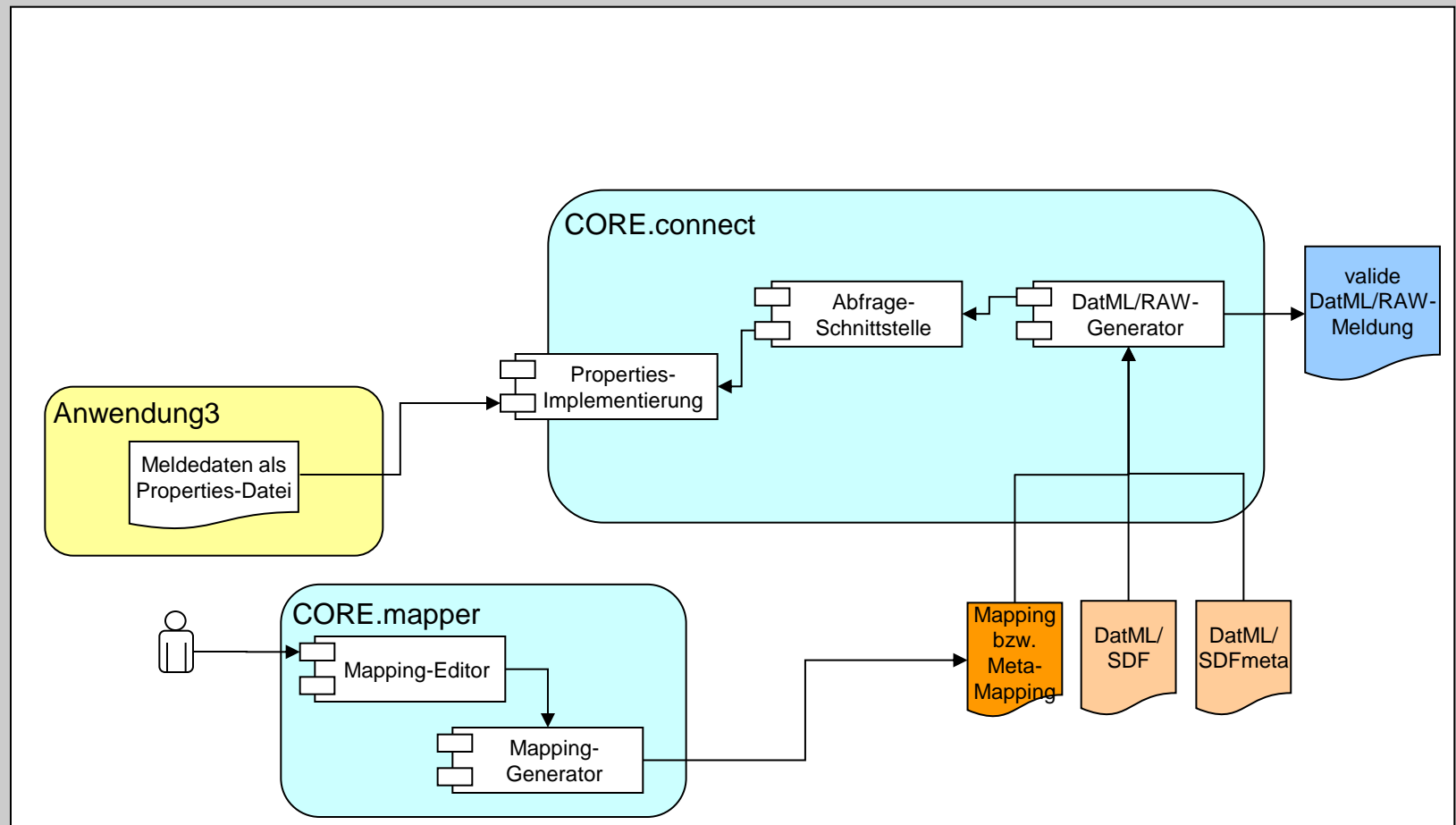
```
12345678;GemeindeName;GeZu;AGS011234;Bemerkungen;...
```

- Es folgen die Zeilen mit den Sätzen der Nachricht

```
1;0;1;GemSchlu;GeZu;GemMeldungNr;12012010;VerdachtsmomenteText;FKS011234;10;01012011;...  
2;1;1;GemSchlu;GeZu;GemMeldungNr;12012010;VerdachtsmomenteText;FKS011234;10;01012011;...  
3;0;1;GemSchlu;GeZu;GemMeldungNr;12012010;VerdachtsmomenteText;FKS011234;10;01012011;...
```

Generisch unterstützte Erzeugung

- **Meldedaten als Properties-Datei**



Generisch unterstützte Erzeugung

- **Meldedaten als Properties-Datei**
 - Properties-Implementierung ermöglicht Erzeugung von DatML/RAW-Meldungen aus Feldname-Wert-Zuordnungstabellen (Properties-Dateien)

```
Lieferungsart=2
Anwendung.Anwendungsname=Testanwendung
Anwendung.Version=10
Anwendung.Hersteller=Testfirma
Absender.Kennung=Kennung
Absender.Organisation=Organisation
Absender.Adresse.Strasse=Test-Str.
Absender.Adresse.Hausnummer=3
Absender.Adresse.Postleitzahl=12345
Absender.Adresse.Ort=Teststadt
Absender.Ansprechpartner.Nachname=Mustermann
Absender.Ansprechpartner.Telefon=0500-12345
Absender.Ansprechpartner.Email=test@test.de
Nachricht[0].ErhebungsID=0004
Nachricht[0].Berichtsempfaenger=00
Nachricht[0].Berichtszeitraum=200701
Nachricht[0].MeldungsID=1
Nachricht[0].Meldungsart=2
Nachricht[0].OrigMeldungsID=1234
Nachricht[0].BerichtseinheitID=BerichtseinheitID
Nachricht[0].GemeindeName=GemeindeName
Nachricht[0].GemeindeZusatz=GemeindeZusatz
Nachricht[0].AGSSchluesselVerzeichnis=AGSSchluesselVerzeichnis
Nachricht[0].Satz[0].MeldungArt=1
Nachricht[0].Satz[0].IstKorrektur=0
...
```

Danke für Ihre Aufmerksamkeit



Achim Mohr
Statistisches Bundesamt, Wiesbaden
achim.mohr@destatis.de

Alf Steilen
Statistisches Bundesamt, Wiesbaden
alf.steilen@destatis.de