

IT-Standards für die amtliche Statistik

## Data Markup Language (DatML)

Komponente DatML/RES 1.0

Ein XML-basierter Dokumenttyp für Prüfprotokolle

Autor: Michael Schäfer, <mailto:michael.schaefer@destatis.de>

Status: Spezifikation

Stand: 09.12.2004

© Statistisches Bundesamt Wiesbaden, Deutschland

# Data Markup Language/RES (DatML/RES) 1.0

## Spezifikation

### **Beschreibung**

Die Data Markup Language (DatML) ist eine Familie XML-basierter Dokumenttypen für statistische Daten und Metadaten. DatML unterstützt den gesamten statistischen Produktionsprozess von der Erhebung der Daten über ihre Plausibilisierung bis zur eigentlichen Verarbeitung und schließlich Archivierung. Den unterschiedlichen Anforderungen im Laufe des Produktionsprozesses entspricht die Unterteilung des Dokumenttyps in eine Reihe von Unterformaten, die untereinander eine konsistente Metadatenhaltung ermöglichen. Einige Komponenten, wie z.B. Datensatzbeschreibungen, werden in mehreren Unterformaten verwendet. Im Einzelnen:

- DatML/RAW für die Lieferung statistischer Rohdaten für beliebige Erhebungen; dieses Format bildet eine abgeschlossene Komponente.
- DatML/RES für das Quittieren von DatML/RAW-Rohdatenlieferungen; dieses Format bildet eine abgeschlossene Komponente.
- DatML/SDF für die Beschreibung statistischer Erhebungen.
- DatML/SSP für die Beschreibung von Datenstrukturen und statistischen Spezifikationen.
- DatML/EDT für die Beschreibung von Plausibilitätsprüfungen.
- DatML/SET für die Beschreibung statistischer Datenbestände ab der Rohdatenübernahme, inkl. Datensatzbeschreibung, Plausibilitätsprüfungen, Umsetzung und Fragebögen, sowie für die sichere, anwendungs- und plattformneutrale Archivierung.
- DatML/ASK für die Beschreibung von Fragebögen.

Im vorliegenden Dokument ist der Dokumenttyp DatML/RES beschrieben.

### **Dokumentstatus**

Dieses Dokument ist die verbindliche Spezifikation des Dokumenttyps DatML/RES in der Version 1.0.

Teile dieses Dokumentes enthalten über die eigentliche Spezifikation hinausgehende Beschreibungen wie z.B. mögliche Implementierungsmodelle und sind daher nicht verbindlich; sie sind in der jeweiligen Überschrift mit der Zeichenfolge *[nicht-normativ]* markiert.

## 1. Inhaltsverzeichnis

1. Inhaltsverzeichnis .....	3
2. Einführung .....	7
2.1. Versionierung, Namensraum, Schemata .....	7
2.2. Unterstützung von DatML/RAW .....	7
2.3. Verwendung .....	8
2.4. Verfügbarkeit .....	8
2.5. Struktur .....	8
2.5.1. Dokument .....	8
2.5.2. Prüfprotokoll .....	9
2.5.2.1. Prüfung .....	9
2.5.2.2. Dokumentebene .....	9
2.5.2.3. Nachrichtenebene .....	9
2.5.2.4. Meldungsebene .....	9
2.6. Metadaten .....	9
2.6.1. Absender und Empfänger .....	10
2.6.2. Berichtspflichtiger und Berichtsempfänger .....	10
2.6.3. Erhebung und Berichtszeitraum .....	10
2.6.4. Verarbeitungsoptionen .....	10
2.6.4.1. Testunterstützung .....	10
2.6.5. Protokoll .....	10
2.6.6. Identifikation des Dokumentes .....	11
2.6.7. Formatierte Werte .....	11
3. Prüfprotokolle .....	12
3.1. Prüfstufen .....	12
3.2. Prüfebene .....	12
3.2.1. Syntax .....	12
3.2.2. Semantik .....	12
3.2.3. Autorisierung .....	13
3.2.4. Daten .....	13
3.2.5. Dokument .....	13
3.3. Dokumentstatus .....	13
3.4. Fehler .....	13
3.4.1. Fehlerschlüssel, -gewicht und -text .....	13
3.4.2. Merkmal .....	13
3.4.3. Position .....	13
3.5. Prüfprotokolle für DatML/RAW .....	14
3.5.1. Prüfstufen .....	14
3.5.2. Zuordnung der Fehler .....	15
3.5.3. Indizierung .....	16
3.6. Prüfprotokolle anderer Dokumenttypen .....	16
4. Elementtypen .....	17
4.1. Darstellung der Elementtypen und Attribute .....	18
4.1.1. Graphische Darstellung der Elementtypen .....	18
4.1.2. Tabellarische Darstellung der Elementtypen .....	18
4.1.3. Tabellarische Darstellung der Attribute .....	19
4.2. Dokument .....	20
4.2.1. Element <Datml-RES-D> .....	20

4.3. Prüfprotokoll .....	21
4.3.1. Element <pruefprotokoll> .....	21
4.3.2. Elemente <austausch>, <berichtigung>, <loeschung>, <nachmeldung>.....	22
4.3.3. Element <autorisierung> .....	23
4.3.4. Element <daten> .....	24
4.3.5. Element <dokument>.....	25
4.3.6. Element <dokumenttyp>.....	26
4.3.7. Element <meldung> .....	27
4.3.8. Element <meldungsart>.....	29
4.3.9. Element <meldungsID>.....	30
4.3.10. Element <nachricht>.....	31
4.3.11. Element <nachrichtenID> .....	32
4.3.12. Elemente <nil>, <original> .....	33
4.3.13. Element <pruefstufe>.....	34
4.3.14. Element <pruefung> .....	35
4.3.15. Element <semantik> .....	36
4.3.16. Element <syntax> .....	37
4.4. Fehler .....	38
4.4.1. Element <fehler> .....	38
4.4.2. Element <gewicht> .....	39
4.4.3. Element <position>.....	40
4.4.4. Element <merkmal> .....	41
4.4.5. Element <schluessel>.....	42
4.4.6. Element <text>.....	43
4.5. Dateneingang .....	44
4.5.1. Element <datei> .....	44
4.5.2. Element <dateneingang> .....	45
4.5.3. Element <dateiname>.....	47
4.5.4. Element <datenformat>.....	48
4.5.5. Element <datentraeger> .....	49
4.5.6. Element <dokumentID>.....	50
4.5.7. Element <eingangsstempel> .....	51
4.5.8. Element <formularID>.....	52
4.5.9. Element <formularname> .....	53
4.5.10. Element <onlineformular> .....	54
4.5.11. Element <papierformular> .....	55
4.5.12. Element <upload> .....	56
4.6. Verfahrensteilnehmer, Identifikation, Kontaktangaben .....	57
4.6.1. Element <absender>.....	57
4.6.2. Element <adresse> .....	58
4.6.3. Element <anrede> .....	59
4.6.4. Element <berichtsempfaenger>.....	60
4.6.5. Element <berichtspflichtiger> .....	61
4.6.6. Element <bundesland>.....	62
4.6.7. Element <email> .....	63

4.6.8. Element <empfaenger>.....	64
4.6.9. Element <externeIdentifikation>.....	65
4.6.10. Element <fax>.....	66
4.6.11. Element <hausnummer>.....	67
4.6.12. Element <identifikation>.....	68
4.6.13. Element <identitaet>.....	69
4.6.14. Element <kontakt>.....	70
4.6.15. Element <kreis>.....	71
4.6.16. Element <land>.....	72
4.6.17. Element <nachname>.....	73
4.6.18. Element <niederlassung>.....	74
4.6.19. Element <organisation>.....	75
4.6.20. Element <ort>.....	76
4.6.21. Element <person>.....	77
4.6.22. Element <postfach>.....	78
4.6.23. Element <postfachleitzahl>.....	79
4.6.24. Element <postfachort>.....	80
4.6.25. Element <postleitzahl>.....	81
4.6.26. Element <strasse>.....	82
4.6.27. Element <telefon>.....	83
4.6.28. Element <titel>.....	84
4.6.29. Element <url>.....	85
4.6.30. Element <vorname>.....	86
4.6.31. Element <zeile>.....	87
4.7. Erhebung, Berichtszeitraum.....	88
4.7.1. Elemente <beginn>, <ende>.....	88
4.7.2. Element <berichtszeitraum>.....	89
4.7.3. Element <erhebung>.....	91
4.7.4. Elemente <jahr>, <halbjahr>, <semester>, <quartal>, <monat>, <woche>, <tag>.....	92
4.8. Verarbeitungssteuerung.....	93
4.8.1. Element <optionen>.....	93
4.8.2. Element <test>.....	94
4.9. Protokollierung.....	95
4.9.1. Übersicht.....	95
4.9.2. Element <anwendung>.....	96
4.9.3. Element <anwendungsname>.....	97
4.9.4. Element <dokumentinstanz>.....	98
4.9.5. Element <hersteller>.....	99
4.9.6. Element <protokoll>.....	100
4.9.7. Element <ressourceID>.....	101
4.9.8. Element <server>.....	102
4.9.9. Element <zertifikat>.....	103
4.10. Sonstige Elemente.....	104
4.10.1. Element <berechtigung>.....	104
4.10.2. Element <datum>.....	105

---

4.10.3. Element <kennung> .....	106
4.10.4. Element <memo>.....	107
4.10.5. Element <name>.....	108
4.10.6. Element <string> .....	109
4.10.7. Element <text>.....	110
4.10.8. Element <uhrzeit> .....	111
4.10.9. Element <version> .....	112
4.10.10. Element <zusatz> .....	113
5. Beispieldokument <i>[nicht-normativ]</i> .....	114
6. XML-Schema 1.0 <i>[nicht-normativ]</i> .....	116

## 2. Einführung

### 2.1. Versionierung, Namensraum, Schemata

Die Versionsbezeichnung für DatML/RES hat das Format:

*version.variante[.korrekturnummer]*

Alle Elemente der Versionsbezeichnung sind ganzzahlige Werte. Wenn sich der Wert eines der Elemente ändert, werden alle rechts davon stehenden Elemente auf den Wert 0 gesetzt. Die Versionsnummer *version* erhöht sich um 1 für jede Version, die mit der Vorgängerversion inkompatibel ist. Die Variante erhöht sich um 1 für jede neue kompatible Version, und die Korrekturnummer für jede Version, die nur Korrekturen enthält. Die Korrekturnummer wird nur bei Bedarf geführt.

Die deutsche und die (geplante) englische Version von DatML/RES werden in den Namen der Schemadateien und den Namensräumen durch einen Sprachcode *lang* mit den Werten *de* oder *en* nach ISO analog dem Attribut `xml:lang` unterschieden.

Die Versionsnummer geht in die Namen der Schemadateien und in die Namensräume ein. Die Variante hat in beiden Fällen stets den Wert 0, und daher ändern sich die Namen der Schemadateien und die Namensräume nicht, solange eine Version kompatibel mit allen Vorgängerversionen ist, welche die gleiche Versionsnummer haben.

#### Namensschemata:

Namensraum	<a href="http://www.destatis.de/schema/datml-res/version.0/lang">http://www.destatis.de/schema/datml-res/version.0/lang</a>
Schema	<a href="#">datml-res-lang-version_0.xsd</a>

Anhand dieser Namensschemata ergeben sich folgende Namensräume und Schemadateinamen

Version	Sprache / Schematyp	Namensraum / Schema
1.0	Deutsch	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>
	Englisch	<a href="http://www.destatis.de/schema/datml-res/1.0/en">http://www.destatis.de/schema/datml-res/1.0/en</a>
	XSD	<a href="#">datml-res-de-1_0.xsd</a> , <a href="#">datml-res-en-1_0.xsd</a>

### 2.2. Unterstützung von DatML/RAW

DatML/RES 1.0 berücksichtigt alle zum Zeitpunkt der Freigabe dieser Version verfügbaren Version von DatML/RAW, also die Versionen 1.0, 1.0.2 und 2.0.

Abhängigkeiten in der Entwicklung der beiden Dokumenttypen sind nicht zwingend. Der Funktionsumfang von DatML/RES ist auch für zukünftige Versionen von DatML/RAW nutzbar, und umgekehrt schränkt eine neue Version von DatML/RES nicht notwendigerweise die Nutzung für die Versionen von DatML/RAW bis 2.0 ein. DatML/RES und DatML/RAW werden daher grundsätzlich unabhängig voneinander entwickelt. Sofern eine zukünftige Version von DatML/RAW eine bestimmte Version von DatML/RES erfordert, muß deren Spezifikation ausdrücklich darauf hinweisen.

### **2.3. Verwendung**

DatML/RES ist komplementär zu DatML/RAW. Mit DatML/RAW werden Rohdaten von den Meldern zu den erhebenden Stellen transportiert, während mit DatML/RES Nachrichten von den erhebenden Stellen an Melder versendet werden. Die primären Szenarien für den Einsatz von DatML/RES in Kombination mit DatML/RAW sind a) der automatisierte Austausch von Rohdaten und Nachrichten zwischen Anwendungen der meldenden und der erhebenden Stellen in einem transaktionsorientierten Verfahren und b) die Übertragung von Informationen an meldende Stellen bei vom Erhebungsvorgang entkoppelten Anforderungen, wie beispielsweise die Abfrage von Prüfprotokollen..

DatML/RES unterstützt in der Version 1.0 die Übertragung von Prüfprotokollen, die Auskunft über die Qualität der eingegangenen Meldungen geben. Eine Meldung muss dabei nicht zwangsläufig im Format DatML/RAW übertragen worden sein; sie kann später nach DatML/RAW konvertiert worden sein oder überhaupt in einem anderen Format vorliegen. Voraussetzung für die Verwendung eines DatML/RES Prüfprotokolls ist, dass die Ergebnisse einer Meldungsprüfung mit DatML/RES abgebildet werden können. Prinzipiell können daher mit DatML/RES die Prüfprotokolle zu beliebigen Meldungsformaten übertragen werden.

### **2.4. Verfügbarkeit**

DatML/RES 1.0 steht nur in einer deutschen Variante zur Verfügung. Für spätere Versionen ist eine englische Variante geplant, die sich lediglich durch die unterschiedliche Benennung der Elemente, Attribute und vordefinierten Attributwerte unterscheidet.

### **2.5. Struktur**

#### **2.5.1. Dokument**

Jedes DatML/RES-Dokument muss folgende Komponenten enthalten: den Absender (<absender>), den Empfänger (<empfaenger>) und genau ein Prüfprotokoll (<pruefprotokoll>).

Optionale Komponenten sind die Verarbeitungsoptionen (<optionen>) und die Protokollinformationen (<protokoll>).

## 2.5.2. Prüfprotokoll

Das Prüfprotokoll besteht aus den Komponenten:

- Prüfung (<pruefung>), mit Angaben zum Prüfverfahren und summarischen Prüfergebnissen für einzelne Prüfungsphasen wie Syntaxprüfung usw.,
- Dokument (<dokument>), mit dem Prüfergebnis für das gesamte Dokument und den auf der Dokumentebene aufgetretenen Fehlern,
- je einer Nachricht (<nachricht>) für jede Nachricht des Meldungsdokumentes mit dem Prüfergebnis für die Nachricht und den auf ihrer Ebene aufgetretenen Fehlern.
- je einer Meldung (<meldung>) für jede Meldung des Meldungsdokumentes mit dem Prüfergebnis für die Meldung und den auf ihrer Ebene aufgetretenen Fehlern.

Dokument-, Nachrichten- und Meldungsebene sind hierarchisch angeordnet.

### 2.5.2.1. Prüfung

Die Angaben zu Prüfverfahren umfassen die Prüfstufe und den verwendeten Dokumenttyp. Die Komponente Prüfung enthält außerdem die summarischen Prüfergebnisse für die Prüfebene Syntax, Semantik, Autorisierung und Daten (siehe 3.2).

### 2.5.2.2. Dokumentebene

Der Dateneingang (<dateneingang>, optional), mit Angaben zum Eingang des Meldungsdokumentes, das Prüfergebnis der Dokumentebene und alle dort festgestellten Fehler sind in <dokument> protokolliert. Für das Dokument wird der Prüfstatus angegeben (*pruefstatus*), und ob es als ganzes *akzeptiert* oder *abgewiesen* wurde (*dokumentstatus*).

### 2.5.2.3. Nachrichtenebene

Die Nachrichtenebene eines Prüfprotokolls enthält außer der optionalen Nachrichten-ID nur die auf dieser Ebene festgestellten Fehler.

### 2.5.2.4. Meldungsebene

In der Meldungsebene befinden sich die meldungsbezogenen Metadaten, insbesondere Erhebung und Berichtszeitraum, sowie optional Berichtspflichtiger und Berichtsempfänger, und die auf der Meldungsebene festgestellten Fehler.

## 2.6. Metadaten

DatML/RES verwendet für die Abbildung der Metadaten die gleichen Strukturen wie DatML/RAW. Für die Befüllung der Strukturen sind daher dessen dokumenttypspezifischen Regeln analog anzuwenden, unter Beachtung einschließlich der weiteren erhebungsspezifischen Vorschriften.

### 2.6.1. Absender und Empfänger

Jedes DatML/RES-Dokument muss genau einen Absender und einen Empfänger angeben. In Bezug auf ein vorausgegangenes Meldungsdocument sind die Rollen von Absender und Empfänger vertauscht. Wurde das Meldungsdocument über einen zentralen Erhebungsserver geschickt, der eine Verteilung vornimmt, und wird das Prüfprotokoll nicht durch jenen Server, sondern erst beim Berichtsempfänger erstellt und dann direkt zum ursprünglichen Absender übertragen, nimmt der Berichtsempfänger die Rolle des Absenders ein.

### 2.6.2. Berichtspflichtiger und Berichtsempfänger

Berichtspflichtiger und Berichtsempfänger werden für jede Meldung separat angegeben. Auf die Angabe kann verzichtet werden, soweit sie mit dem Absender bzw. dem Empfänger des DatML/RES-Dokumentes übereinstimmen.

### 2.6.3. Erhebung und Berichtszeitraum

Erhebung und Berichtszeitraum sind für jede Meldung vollständig anzugeben.

### 2.6.4. Verarbeitungsoptionen

Verarbeitungsoptionen geben dem Absender eines DatML/RES-Dokumentes die Möglichkeit, die Verarbeitung des Dokumentes in bestimmter Weise zu beeinflussen und Aktionen anzufordern bzw. auszulösen. Es ist der jeweiligen Anwendung überlassen, diese Funktionen tatsächlich zu implementieren und zur Verfügung zu stellen. Jede Option ist ein nachgeordnetes Element des Top-Level-Knotens `<optionen>`. DatML/RES unterstützt als einzige Verarbeitungsfunktion die Kennzeichnung von Dokumenten als Testdokument (`<test>`).

#### 2.6.4.1. Testunterstützung

Der Elementtyp `<test>` zeigt an, dass eine Datenlieferung allein zu Testzwecken erfolgt. Das Attribut `kennung` dient zur Unterscheidung verschiedener Testläufe, Dokumentvarianten etc.

### 2.6.5. Protokoll

Protokollinformationen (`<protokoll>`) dienen dazu, die Instanzen und den Verarbeitungsweg eines DatML/RES-Dokumentes zu dokumentieren. Jede Dokumentinstanz (`<dokumentinstanz>`) kann separat protokolliert werden.

Protokollinformationen sind optional; für die Befüllung der Protokollstrukturen gilt folgende Regel:

**Regel:**

- Dokumentinstanzen sind in der Reihenfolge ihrer Entstehung zu protokollieren, d.h. in einer Folge von Elementen des Typs `<dokumentinstanz>` beschreibt das erste die ursprüngliche und das letzte die aktuelle Instanz des Dokumentes.

- Wird ein DatML/RES-Dokument gelesen und daraus eine neue Dokumentinstanz erzeugt, sind dessen Dokumentinstanzeinträge zu übernehmen.
- Alle in einem Element `<dokumentinstanz>` enthaltenen Informationen beziehen sich auf das jeweils aktuelle Dokument und die Anwendung, die es erstellt hat.
- Die Dokumentinstanzeinträge sind absteigend nach Alter abzulegen, sodass der letzte Eintrag das aktuelle Dokument beschreibt.

#### **2.6.6. Identifikation des Dokumentes**

Es ist möglich, für jede Instanz eines DatML/RES-Dokumentes eine Identifizierungskennung zu vergeben. Die Identifizierungskennung wird im Element `<dokumentID>` der Elemente `<dokumentinstanz>` und `<datei>` abgelegt. Sie ist für externe wie interne Zwecke verwendbar. Das Format der Identifizierungskennung ist anwendungsabhängig und darf die Wohlgeformtheit des Dokumentes nicht beeinträchtigen.

#### **2.6.7. Formatierte Werte**

Mit dem Element `<string>` ist es möglich, formatierte Zeichenketten zu verwenden. Die erforderlichen Attribute `format` und `klasse` liefern ein Formatschema und die Klasse des Formatschemas. Aufgrund des Klassennamens kann das Formatschema interpretiert werden.

Klassen und Formatschemata sind detailliert in der jeweiligen Spezifikation des Dokumenttyps DatML/RAW beschrieben.

### **3. Prüfprotokolle**

Ein Prüfprotokoll (<pruefprotokoll>) enthält die Prüfergebnisse für ein Meldungsdocument, unterteilt nach drei Ebenen: Dokument (<dokument>), Nachricht (<nachricht>) und Meldung (<meldung>). Diese Unterteilung entspricht der *logischen* Unterteilung eines DatML/RAW-Dokumentes.

Der Umfang der Prüfungen und damit des Prüfprotokolls kann in Abhängigkeit vom Meldungsformat und den Prüfverfahren variieren und ist für diese entsprechend zu dokumentieren.

#### **3.1. Prüfstufen**

In vielen Fällen ist es aus verfahrenstechnischen oder fachlichen Gründen nicht möglich, ein Meldungsdocument vollständig in einem einzigen Schritt zu prüfen. Stattdessen werden zwei oder mehr Prüfungen durchlaufen, die jeweils unterschiedliche Aspekte eines Dokumentes prüfen oder in der Tiefe variieren. Prüfprotokolle können daher Zwischenergebnisse enthalten, und es kann vorkommen, dass ein Dokument so fehlerhaft ist, dass es nicht alle Prüfschritte durchlaufen kann. Es ist daher sinnvoll, in einem Prüfprotokoll eine Information zu hinterlegen, die den Umfang der vorgenommenen Prüfungen ausdrückt; dies geschieht in DatML/RES durch die Angabe einer Prüfstufe (<pruefstufe>).

Für die Prüfungen von DatML/RAW-Dokumenten wurden die unter 3.5 beschriebenen Prüfstufen definiert.

#### **3.2. Prüfebene**

DatML/RES kennt fünf generische Prüfebene, die verschiedenen formalen und inhaltlichen Aspekten entsprechen, bezüglich derer ein Meldungsdocument geprüft werden kann. Für jede Prüfebene – Syntax (<syntax>), Semantik (<semantik>), Autorisierung (<autorisierung>), Daten (<daten>) und Dokument (<dokument>) – kann angegeben werden, ob sie *ungeprüft*, *fehlerfrei* oder *fehlerhaft* ist. Dieses summarische Prüfergebnis wird als Prüfstatus (Attribut `pruefstatus`) bezeichnet.

##### **3.2.1. Syntax**

Diese Prüfebene entspricht einer reinen Syntaxprüfung. Ein XML-basiertes Dokument ist syntaktisch korrekt, wenn es den XML-Syntaxregeln folgt.

##### **3.2.2. Semantik**

Diese Prüfebene entspricht einer reinen Semantikprüfung. Ein XML-Dokument ist aus XML-Sicht semantisch korrekt, wenn es gegen eine Dokumenttypdefinition validierbar ist. Es kann aber semantische Regeln geben, die nicht durch einen Dokumenttyp dargestellt werden können. Dann tritt zur semantischen Prüfung auf der Basis der Dokumenttypdefinition die Prüfung dieser Regeln durch eine dokumenttyp-

spezifische Anwendung. In den Bereich der Semantikprüfung fallen auch solche Daten, die formale Aspekte der Meldung betreffen oder rein funktionaler, nicht-statistischer Art sind, z.B. Kontrollwerte wie die Anzahl der Datensätze oder Testindikatoren.

### 3.2.3. Autorisierung

Diese Prüfebene umfasst die Prüfung der Angaben zur Identifikation und Autorisierung der Verfahrensteilnehmer, z.B. Kennungen und Adressen.

### 3.2.4. Daten

Gegenstand der Prüfebene Daten sind alle *statistischen* Rohdaten und Metadaten.

### 3.2.5. Dokument

Die Prüfebene Dokument fasst alle anderen Prüfebeneen zusammen. In ihr wird zusätzlich mit dem Dokumentstatus (Attribut `dokumentstatus`) angegeben, ob das Dokument als ganzes *ungeprüft* ist, *akzeptiert* oder *abgewiesen* wurde. Wie Prüfstatus und Dokumentstatus korrelieren ist verfahrensspezifisch und nicht durch diese Spezifikation geregelt.

## 3.3. Dokumentstatus

Siehe 3.2.5.

## 3.4. Fehler

Ein erkannter Fehler gemäß der für den jeweiligen Dokumenttyp geltenden Vorgaben (z.B. für DatML/RAW, siehe 3.5.2) einer der Prüfprotokollebenen (3.2) zuzuordnen und dort zu beschreiben.

### 3.4.1. Fehlerschlüssel, -gewicht und -text

Ein Fehler kann mit Schlüssel, Gewicht und Text beschrieben werden. Welche dieser Elemente verwendet zur Fehlerbeschreibung werden ist verfahrensspezifisch festzulegen und nicht durch diese Spezifikation geregelt. Für den gesamten Fehler, den Schlüssel, das Gewicht und den Text bietet jeweils das Attribut `klasse` die Möglichkeit, die Inhalte einer Klassifikation oder sonstigen formalen Definition zuzuordnen.

### 3.4.2. Merkmal

Bezieht sich der Fehler auf ein Merkmal, sollte dieses nach Möglichkeit in der Fehlerbeschreibung (`<merkmal>`) enthalten sein..

### 3.4.3. Position

Die Position (`<position>`) eines Fehlers ist alternativ als XPath-Ausdruck (Default), als absoluter Byte-Index, als absoluter Zeichenindex, als Satz- und Zeichenindex oder als Bezeichner – z.B. eines Formularfeldes – darstellbar. Alle Indizes beginnen mit 1.

format	Bedeutung und Beispiel
xpath	<p>Ein Xpath-Ausdruck, der auf das Element oder das Attribut verweist, das den Fehler verursacht; wie präzise dieser Verweis ist, ob er absolut oder relativ ist, und wie z.B. multiple Elemente indiziert werden, ist anwendungsabhängig. Beispiele:</p> <pre> &lt;position&gt;/DatML-RAW-D/absender/kennung&lt;/position&gt; &lt;position&gt;/DatML-RAW-D/@version&lt;/position&gt; &lt;position&gt;satz/mm[5]/wert&lt;/position&gt; &lt;position&gt;satz/mm[@name='Alter']/wert&lt;/position&gt; &lt;position&gt;mmgr[@name='Haushalt'][1]/mm[4]&lt;/position&gt; &lt;position&gt;mmgr[34]/mm[12]/wert/@faktor&lt;/position&gt; </pre>
byte	<p>Der absolute dezimale Index eines Bytes; Beispiel:</p> <pre> &lt;position format='byte'&gt;2691&lt;/position&gt; </pre>
char	<p>Der absolute dezimale Index eines Zeichens; Beispiel:</p> <pre> &lt;position format='char'&gt;2691&lt;/position&gt; </pre>
name	<p>Ein Bezeichner, z.B. der Name oder Beschreibungstext eines Formularfeldes:</p> <pre> &lt;position format='name'&gt;Alter in Jahren&lt;/position&gt; </pre>
satz	<p>Der Index <i>c</i> eines Zeichens im Satz mit dem Index <i>s</i> in der Form <i>s, c</i>; Beispiel Satz 8, Zeichen 25:</p> <pre> &lt;position format='satz'&gt;8,25&lt;/position&gt; </pre>

### 3.5. Prüfprotokolle für DatML/RAW

Prüfprotokolle reflektieren nicht die Segmentierung von DatML/RAW-Nachrichten. Die zu einer Meldung gehörenden Metadaten (Erhebung, Berichtszeitraum usw.) werden ungeachtet möglicher Redundanzen vollständig bei der Meldung abgelegt.

#### 3.5.1. Prüfstufen

Für DatML/RAW wurden fünf Prüfstufen definiert, denen ein Wert von 1 bis 5 entspricht:

#	Umfang der Prüfungen	Enthält Stufe
---	----------------------	---------------

1	Konformität mit der Dokumenttypdefinition (DTD oder XML-Schema), also Wohlgeformtheit (Syntax) und Validität (Semantik).	-
2	Semantik, soweit sie nicht durch die Dokumenttypdefinition abgedeckt ist, z.B. Eindeutigkeit der Metadaten im Segmentpfad.	1
3	Einhaltung weiterer, nicht statistikspezifischer Vorgaben, wie Gültigkeit der Empfänger- und Absenderangaben.	1,2
4	Einhaltung der Vorgaben der Erhebungsbeschreibung, z.B. Namen und Anzahl der Merkmale und Merkmalsgruppen, Vollständigkeit der obligatorischen Merkmale, Datentypen, Berichtszeitraum und Erhebung.	1,2
5	Vollständige Prüfung	1,2,3,4

### 3.5.2. Zuordnung der Fehler

Die Regel für die Zuordnung eines in einem DatML/RAW-Dokument festgestellten Fehlers zu den logischen Ebenen eines Prüfprotokolls lautet:

**Regel:** Ein bei der Prüfung eines DatML/RAW-Dokumentes festgestellter Fehler ist wie folgt den logischen Ebenen des Prüfprotokolls zuzuordnen:

- 1) Der *aktuellen Meldung*: Wenn der Fehler nach Erreichen eines Datensegmentknotens (<datensegment>) festgestellt wird, und die fehlerverursachende Komponente logisch zu der Meldung gehört, also zum Beispiel auch dann, wenn es sich bei dieser Komponente um ein dem Datensegment übergeordnetes Metadatum der Meldung handelt;
- 2) Der *aktuellen Nachricht*: Wenn der Fehler nach Erreichen eines Nachrichtenknotens (<nachricht>) und – falls die Nachricht Segmentknoten (<segment>) enthält – vor Erreichen eines Segmentknotens festgestellt wird, und die fehlerverursachende Komponente logisch zu der Nachricht gehört; ist die Nachricht unsegmentiert, ist der Fehler der Meldung zuzuordnen, wenn die fehlerverursachende Komponente logisch zur Meldung gehört.
- 3) Der *Dokumentebene*:
  - a) wenn es sich um einen Fehler handelt, der die Verarbeitung des Dokumentes im technischen Sinn verhindert, insbesondere um Fehler in der Wohlgeformtheit (Syntax) oder Validität (Semantik) gemäß der [XML-Spezifikation des World Wide Web Consortiums](#).
  - b) wenn der Fehler vor Erreichen der ersten Nachricht in einem Dokument auftritt, zum Beispiel ein Autorisierungsfehler wie eine ungültige Absenderkennung.
  - c) In allen Fällen, in denen ein Fehler nicht eindeutig einem definierten Abschnitt des Dokumentes zugeordnet werden kann, auch dann, wenn dies allein aus technischen Gründen nicht geschieht.

### 3.5.3. Indizierung

Es ist notwendig, Elemente, die im Inhalt des übergeordneten Elementes mehr als einmal vorkommen können, bei der Konstruktion eines Xpath-Ausdrucks durch die Angabe eines Index präzise zu adressieren. Die wichtigsten dieser Elemente sind `<nachricht>`, `<segment>`, `<mmgr>` und `<mm>`; die beiden zuletzt genannten müssen außerdem ein Attribut `name` haben, das zur Adressierung verwendet werden kann, bei Merkmalsgruppen zusätzlich zum Index und bei Merkmalen alternativ zum Index.

Es wird empfohlen, ausschließlich den Index zu verwenden, da dieser immer eindeutig ist und bei Parsen ermittelt wird und daher nicht vom Fehlen oder Vorhandensein einer Information im Dokument beeinflusst wird.

### 3.6. Prüfprotokolle anderer Dokumenttypen

Die Beschreibung von Prüfprotokollen anderer Dokumenttypen ist nicht Gegenstand dieser Spezifikation. Es wird empfohlen, sich beim Einsatz für andere Dokumenttypen an folgenden allgemeinen Leitlinien zu orientieren:

- a) Hat ein Dokumenttyp keine einer Nachrichtenebene entsprechende Struktur, kann die Nachrichtenebene im Prüfprotokoll transparent behandelt werden.
- b) Es muss möglich sein, eine ausreichende Menge der verfügbaren Metadaten sinngemäß auf die DatML/RES-Strukturen abzubilden.
- c) Der Dokumenttyp der Meldung sollte im Dateneingang (`<dateneingang>`) dokumentiert werden.

## **4. Elementtypen**

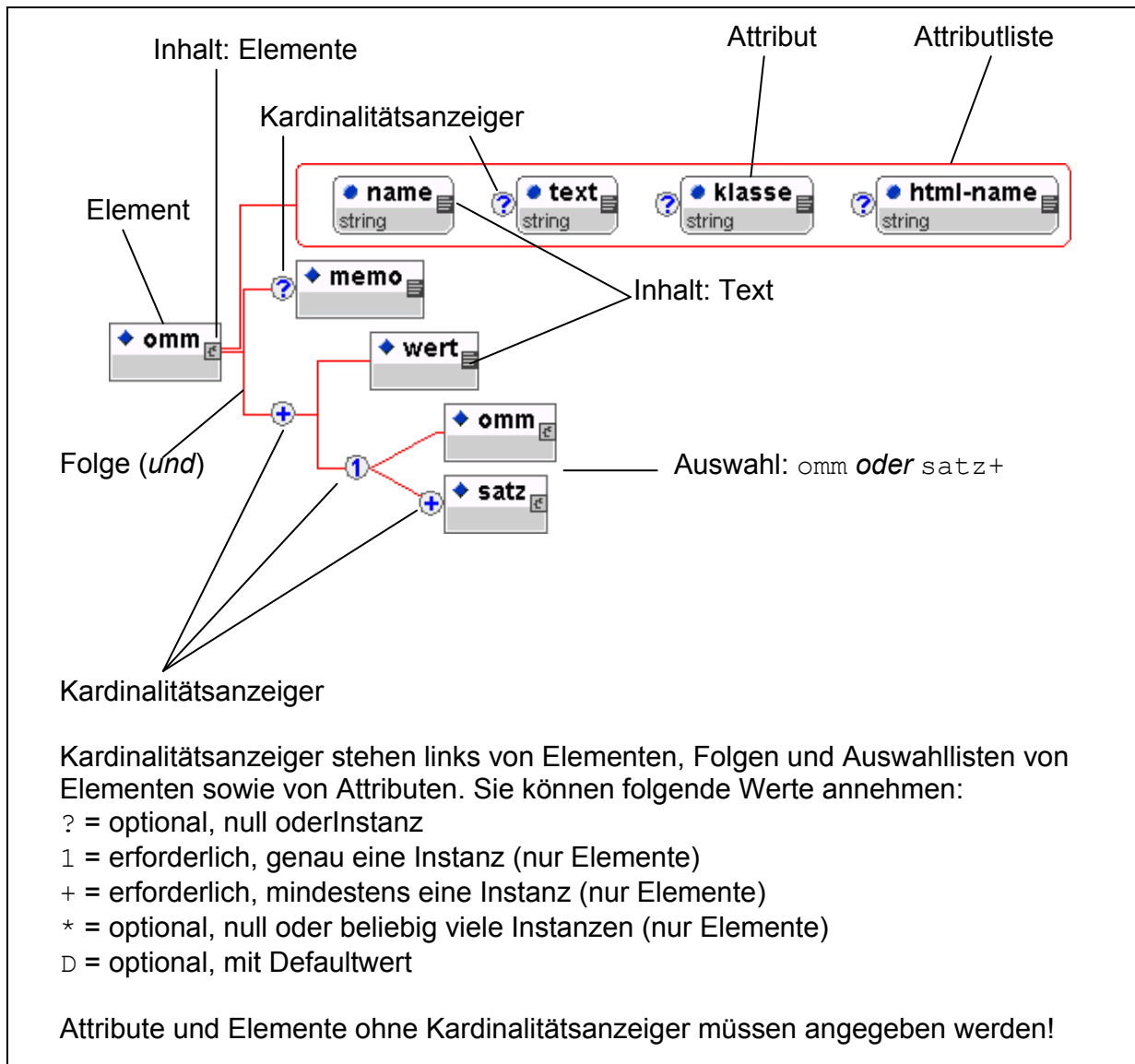
Elementtypen sind in der Beschreibung nach funktionalen und strukturellen Kriterien gruppiert und innerhalb jeder Gruppe alphabetisch aufsteigend sortiert; Attribute sind mit ihren Elementtypen beschrieben:

4.1 Darstellung der Elementtypen und Attribute

4.2 Dokument

## 4.1. Darstellung der Elementtypen und Attribute

### 4.1.1. Graphische Darstellung der Elementtypen



### 4.1.2. Tabellarische Darstellung der Elementtypen

<b>Element</b>	<i>elementname</i>	
Status	<i>status</i>	[r] <i>version</i>
Datentyp:	<i>datentyp</i>	
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de	
<i>graphisches Inhaltsmodell (s. 4.1.1)</i>		

Die Zeile Status erscheint nur bei Bedarf. *status* kann die Werte **veraltet** und **neu** annehmen. *version* legt den Gültigkeitsraum für den angegebenen Status fest, wobei der optionale relationale Operator *r* mit den Werten ' $\leq$ ' und ' $\geq$ ' den Gültigkeitsraum mit einem einschließenden ‚ab‘ bzw. ‚bis‘ genauer eingrenzt.

#### 4.1.3. Tabellarische Darstellung der Attribute

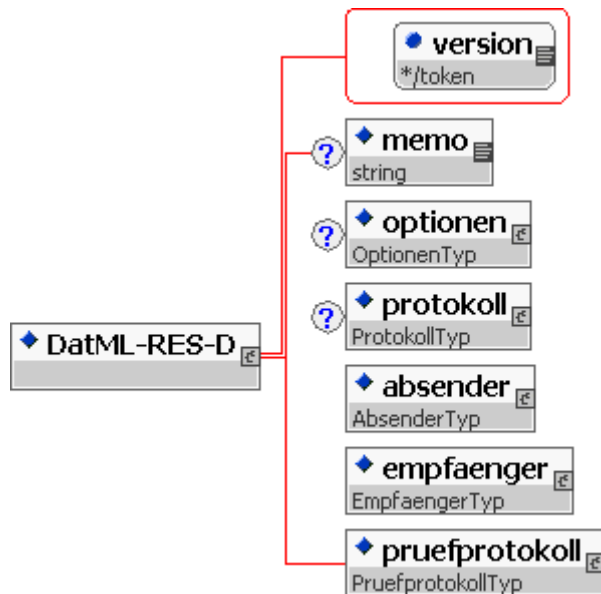
Die Attribute eines Elementtyps sind in einer eigenen Tabelle mit folgenden Spalten beschrieben:

Spalte	Bedeutung	
Attribute	Die Namen der Attribute	
S	Spezifikationsart des jeweiligen Attributes: ? = optional D = optional, mit Defaultwert M = erforderlich (Muss-Angabe)	
Datentyp...	Datentyp, Defaultwert und Werteraum des Attributes	
	Datentyp	Beschreibung
	<i>char[n..m]</i>	Zeichen [zulässige Länge]
	<i>int[n..m]</i>	Ganzzahlen [Wertebereich]
	<i>enumeration</i>	Eine Liste vordefinierter Werte

## 4.2. Dokument

### 4.2.1. Element <Datml-RES-D>

<b>Element</b>	<Datml-RES-D>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



<b>Attribute:</b>	<b>S</b> <b>Datentyp, Default, Werte</b>
version	M <i>enumeration:1.0</i> ; Versionsbezeichnung

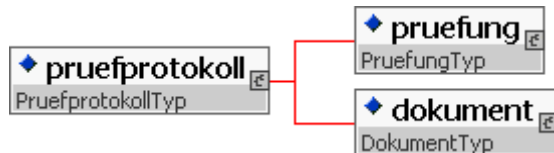
#### Details

Eine Datenlieferung enthält genau eine Absenderangabe, genau eine Empfängerangabe und genau ein Prüfprotokoll. Optional sind ein Kommentar (memo), Verarbeitungsoptionen und Protokoll Daten (Dokumentation der Dokumentinstanzen). Der Teilname '-D' zeigt die deutsche Version des Dokumenttyps an.

## 4.3. Prüfprotokoll

### 4.3.1. Element <pruefprotokoll>

<b>Element</b>	<pruefprotokoll>
Datentyp:	<i>element-content</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

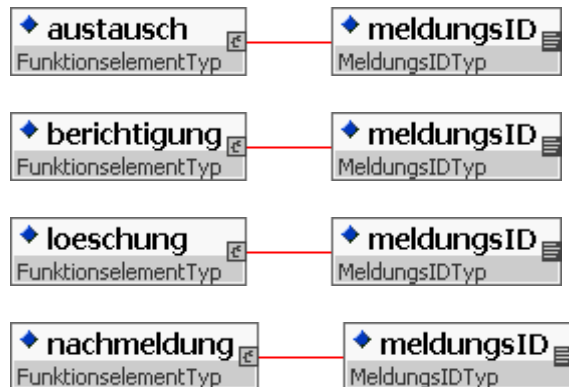
Das Element <pruefprotokoll> enthält das vollständige Prüfprotokoll, unterteilt in die Beschreibung der Prüfung und der summarischen Ergebnisse (<pruefung>) und die Dokumentebene (<dokument>). Die Nachrichtenebene und die Meldungsebene sind der Dokumentebene nachgeordnet.

### 4.3.2. Elemente <austausch>, <berichtigung>, <loeschung>, <nachmeldung>

**Element** <austausch>, <berichtigung>, <loeschung>, <nachmeldung>

Datentyp: *element-content*

Namespace: <http://www.destatis.de/schema/datml-res/2.0/de>



#### Details

Funktionselemente mit Bezug auf eine vorausgegangene Meldung (4.3.8).

### 4.3.3. Element <autorisierung>

<b>Element</b>	<autorisierung>
Datentyp:	<i>empty-element</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



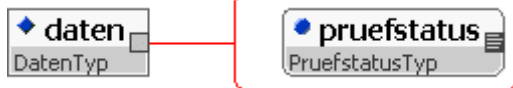
**Attribute:**        **S** **Datentyp, Default, Werte**  
pruefstatus    **M** *enumeration*: ungeprueft | fehlerfrei | fehlerhaft

#### Details

<autorisierung> enthält den Prüfstatus der Prüfebene: Autorisierung.

#### 4.3.4. Element <daten>

<b>Element</b>	<daten>
Datentyp:	<i>empty-element</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



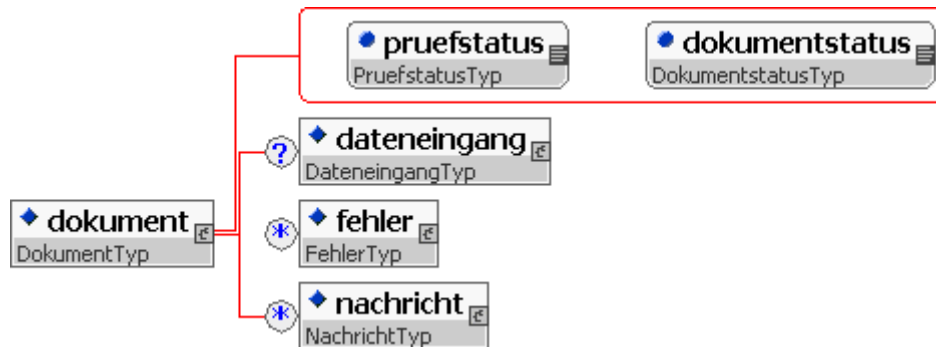
**Attribute:**     **S** **Datentyp, Default, Werte**  
pruefstatus   **M** *enumeration*: ungeprueft | fehlerfrei | fehlerhaft

#### Details

<daten> enthält den Prüfstatus der Prüfebene: Daten.

### 4.3.5. Element <dokument>

<b>Element</b>	<dokument>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Attribute: **S** Datentyp, Default, Werte

pruefstatus	M	<i>enumeration</i> :ungeprueft fehlerfrei fehlerhaft
dokumentstatus	M	<i>enumeration</i> :ungeprueft akzeptiert abgewiesen

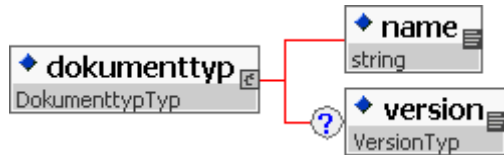
#### Details

<dokument> protokolliert die Ergebnisse der Prüfung der Dokumentenebene. Hier sind der Prüfstatus, also das summarischen Prüfergebnis der Dokumentenebene, und der Dokumentstatus angegeben, der Auskunft darüber gibt, ob das Dokument akzeptiert ist oder nicht. Weitere Inhalte der Dokumentenebene sind die Beschreibung des Dateneingangs, also des Eintreffens des Meldungsdocumentes, und die auf der Dokumentenebene festgestellten Fehler.

Die Nachrichtenebene ist der Dokumentenebene nachgeordnet und wird durch eine Liste von <nachricht>-Elementen dargestellt. Die Liste kann leer sein, z.B. wenn die Prüfung wegen schwerer Fehler bereits auf der Dokumentenebene abgebrochen wurde.

### 4.3.6. Element <dokumenttyp>

<b>Element</b>	<dokumenttyp>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de

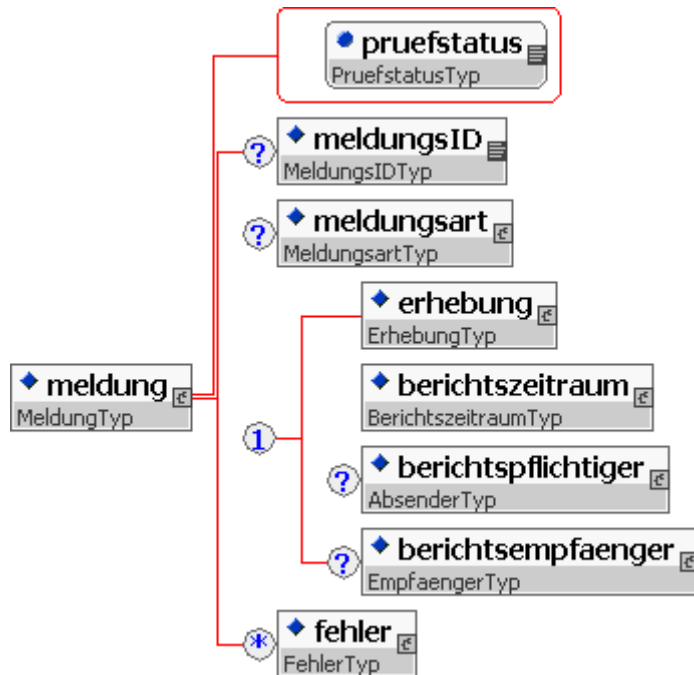


#### Details

<dokumenttyp> gibt den Namen und optional die Version des zur Prüfung verwendeten Dokumenttyps an.

### 4.3.7. Element <meldung>

<b>Element</b>	<meldung>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



**Attribute:**        **S** **Datentyp, Default, Werte**  
 pruefstatus    **M** *enumeration*:ungeprueft|fehlerfrei|fehlerhaft

#### Details

<meldung> protokolliert die Prüfung der Meldung(en) des Meldungsdocumentes. Werden die Meldungen im Meldungsdocument mit einer Kennung o.ä. eindeutig identifizierbar gemacht, ist diese Kennung zu übernehmen und im Element <meldungsID> abzulegen. Die Meldungsart gibt den Verwendungszweck der Meldung an, z.B. Korrektur oder Löschung.

Erhebung und Berichtszeitraum sind vollständig anzugeben, Berichtspflichtiger und -empfänger soweit sie nicht mit dem Empfänger bzw. dem Absender des Prüfprotokolls identisch sind.

Alle gleichnamigen Elemente können aus einem DatML/RAW-Dokument kopiert werden.

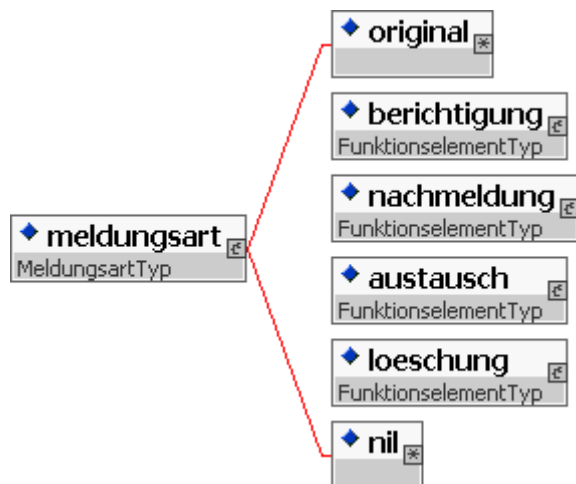
Beispiel:

```
<meldung pruefstatus='fehlerhaft'>
```

```
<erhebung>
  <kennung klasse='ERHID'>51111.1.200204.1</kennung>
</erhebung>
<berichtszeitraum>
  <jahr>2001</jahr>
  <monat>01</monat>
</berichtszeitraum>
<fehler>
  <schluessel>10299</schluessel>
  <text>Wert nicht numerisch</text>
  <position>nachricht[1]/satz[27]/mmgr[8]/mm[19]/wert</position>
</fehler>
</meldung>
```

### 4.3.8. Element <meldungsart>

<b>Element</b>	<meldungsart>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

Die Meldungsart drückt aus, für welchen Zweck die Meldung erfolgte, für die das Prüfprotokoll erstellt wurde. Aus einem DatML/RAW-Dokument kann dieses Element unverändert übernommen werden.

Der Zweck einer Meldung wird durch ein dem Element <meldungsart> nachgeordnetes sogenanntes Funktionselement ausgedrückt. Ist ein Bezug zu einer vorausgegangenen Meldung notwendig oder möglich, erfolgt dieser in Form einer Meldungs-ID, die wiederum dem Funktionselement nachgeordnet ist. Das Funktionselement <nil> unterdrückt die Verarbeitung einer Meldung.

### 4.3.9. Element <meldungsID>

<b>Element</b>	<meldungsID>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



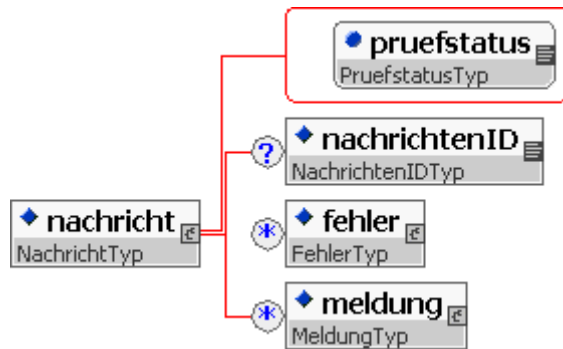
<b>Attribute:</b>	<b>S</b> <b>Datentyp, <u>Default</u>, Werte</b>
<code>klasse</code>	? <i>char</i> ; Klasse oder Klassifikation

#### Details

Mit dem Element <meldungsID> kann einer Meldung ein Identifikator zugeordnet werden. Das Element erscheint in einem Datensegment, weil dies der Knoten ist, an dem eine Meldung in der Segmenthierarchie eindeutig ist.

### 4.3.10. Element <nachricht>

<b>Element</b>	<nachricht>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



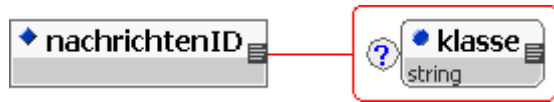
**Attribute:**     **S** Datentyp, Default, Werte  
 pruefstatus    **M** *enumeration*: ungeprueft | fehlerfrei | fehlerhaft

#### Details

<nachricht> protokolliert die Prüfung der Nachrichten(ebene) des Meldungsdocumentes. Werden die Nachrichten im Meldungsdocument mit einer Kennung o.ä. eindeutig identifizierbar gemacht, ist diese Kennung zu übernehmen und im Element <nachrichtenID> abzulegen.

### 4.3.11. Element <nachrichtenID>

<b>Element</b>	<nachrichtenID>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



<b>Attribute:</b>	<b>S</b> <b>Datentyp, <u>Default</u>, Werte</b>
klasse	? <i>char</i> ; Klasse oder Klassifikation

#### Details

Mit dem Element <nachrichtenID> kann einer Nachricht ein Identifikator zugeordnet werden.

### 4.3.12. Elemente <nil>, <original>

<b>Element</b>	<nil>, <original>
Datentyp:	<i>empty-element</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

Funktionselemente *ohne* Bezug auf eine vorausgegangene Meldung (4.3.8).

### 4.3.13. Element <pruefstufe>

<b>Element</b>	<pruefstufe>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



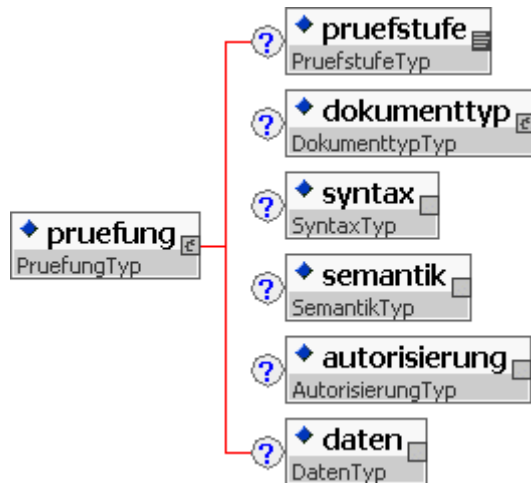
<b>Attribute:</b>	<b>S</b> <b>Datentyp, <u>Default</u>, Werte</b>
klasse	? <i>char</i> ; Optionale Klasse oder Klassifikation

#### Details

<pruefstufe> gibt – falls anwendbar – die Prüfstufe an, die das Prüfverfahren bis zur Erstellung des Prüfprotokolls durchlaufen hat.

#### 4.3.14. Element <pruefung>

<b>Element</b>	<pruefung>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

<pruefung> beschreibt die Prüfparameter und die summarischen Prüfergebnisse.

### 4.3.15. Element `<semantik>`

<b>Element</b>	<code>&lt;semantik&gt;</code>
Datentyp:	<i>empty-element</i>
Namespace:	<code>http://www.destatis.de/schema/datml-res/1.0/de</code>



**Attribute:**     **S** **Datentyp, Default, Werte**  
`pruefstatus`   **M** *enumeration*:`ungeprueft|fehlerfrei|fehlerhaft`

#### Details

`<semantik>` enthält den Prüfstatus der Prüfebene: Semantik.

### 4.3.16. Element <syntax>

<b>Element</b>	<syntax>
Datentyp:	<i>empty-element</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



**Attribute:**     **S** **Datentyp, Default, Werte**  
pruefstatus    **M** *enumeration*: ungeprueft | fehlerfrei | fehlerhaft

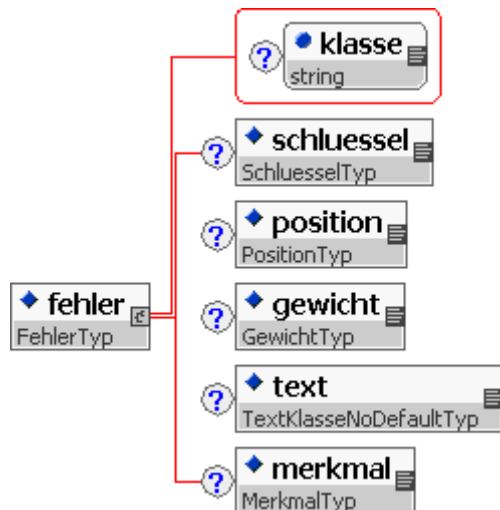
#### Details

<syntax> enthält den Prüfstatus der Prüfebene: Syntax.

## 4.4. Fehler

### 4.4.1. Element <fehler>

<b>Element</b>	<fehler>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



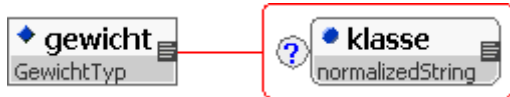
<b>Attribute:</b>	<b>S</b> Datentyp, <u>Default</u> , Werte
klasse	? <i>char</i> ; Klasse oder Klassifikation

#### Details

<fehler> beschreibt einen Fehler mit Schlüssel, Gewicht Text und Position und ggf. dem Merkmal, auf das er sich bezieht. Die Verwendung dieser Elemente und ihre Inhalte sind verfahrensspezifisch festzulegen

#### 4.4.2. Element <gewicht>

<b>Element</b>	<gewicht>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



<b>Attribute:</b>	<b>S</b> <b>Datentyp, <u>Default</u>, Werte</b>
klasse	? <i>char</i> ; Klasse oder Klassifikation

#### Details

Ein Fehlergewicht; die zulässigen Werte sind verfahrensspezifisch.

### 4.4.3. Element <position>

<b>Element</b>	<position>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



<b>Attribute:</b>	<b>S</b> <b>Datentyp, <u>Default</u>, Werte</b>
format	M <i>enumeration</i> :xpath byte char satz feld

#### Details

Die Position der fehlerverursachenden Stelle im Meldungsdokument als Xpath-Ausdruck, Byte-, Zeichen- oder Satz-Offset oder als Name, z.B. eines Formularfeldes. Siehe 3.4.1.

#### 4.4.4. Element `<merkmal>`

<b>Element</b>	<code>&lt;merkmal&gt;</code>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

Der Name eines Merkmals.

#### 4.4.5. Element <schluessel>

<b>Element</b>	<schluessel>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



<b>Attribute:</b>	<b>S</b> <b>Datentyp, <u>Default</u>, Werte</b>
klasse	? <i>char</i> ; Klasse oder Klassifikation

#### Details

Ein Fehlerschlüssel; die zulässigen Werte sind verfahrensspezifisch.

#### 4.4.6. Element <text>

**Element** <text>

Datentyp: *char*

Namespace: <http://www.destatis.de/schema/datml-res/1.0/de>



**Attribute:** **S** **Datentyp, Default, Werte**

klasse ? *char*; Klasse oder Klassifikation

#### Details

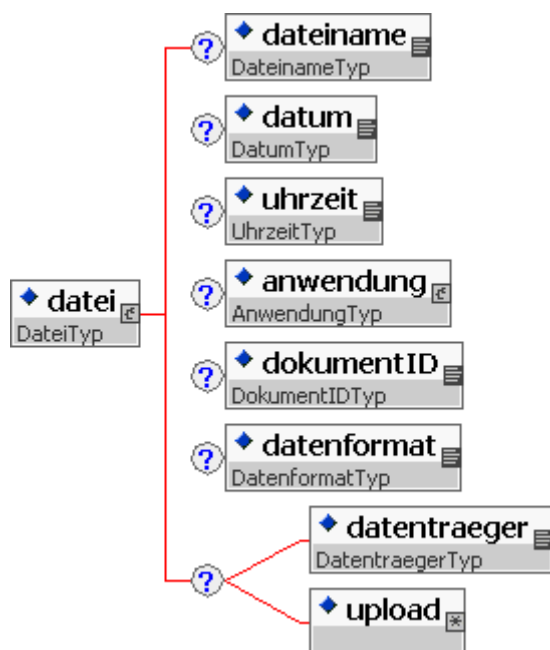
Enthält die textuelle Beschreibung eines Fehlers; die Fehlertexte sind verfahrensspezifisch festzulegen.

## 4.5. Dateneingang

In DatML/RES ist der `<dateneingang>` nicht im Element `<protokoll>` enthalten. Der Dateneingang wird nur als historischer Wert für das Meldedokument abgelegt und ist Bestandteil des Elements `<dokument>`.

### 4.5.1. Element `<datei>`

<b>Element</b>	<code>&lt;datei&gt;</code>
Datentyp:	<i>empty-element</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



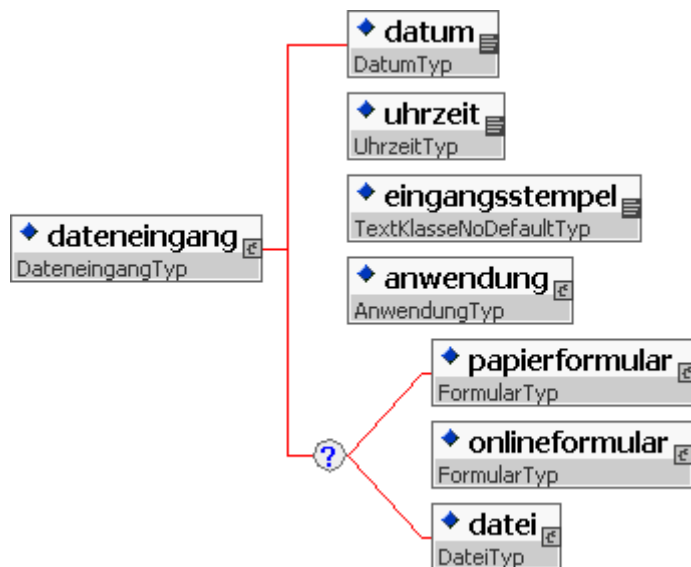
#### Details

`<datei>` dokumentiert die Lieferung der Daten in Dateiform. Es können die Datei selbst (`<dateiname>`), Datum und Uhrzeit ihrer Erzeugung, die Anwendung, welche die Lieferdatei erzeugte (`<anwendung>`), sofern bekannt, ein mit der Datei gelieferter Identifikator (`<dokumentID>`), das Datenformat (`<datenformat>`) und der Lieferweg (`<datentraeger>` bzw. `<upload>`) beschrieben werden.

Die Lieferung einer DatML/RES-Instanz wird im Element `<datenformat>` durch Eintrag des Wertes `datml` im Attribut `datenformattyp` dokumentiert. Enthält die DatML/RES-Instanz ein Element `<dokumentinstanz>` (s. 4.9.4), dann werden – sofern vorhanden – dessen Kindelemente `<dateiname>`, `<datum>`, `<uhrzeit>`, `<dokumentID>` und `<anwendung>` in das Element `<datei>` übernommen.

#### 4.5.2. Element <dateneingang>

<b>Element</b>	<dateneingang>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

<dateneingang> ermöglicht die Dokumentierung von Details des Dateneingangs. Die Erzeugung der Dokumentinstanz und die Datenlieferung können verbundene oder getrennte Aktionen sein, d.h. das Lieferformat wurde entweder vor Lieferung beim Absender oder nach Lieferung beim Empfänger – z.B. bei Online-Erhebungen – erzeugt.

Es können Datum und Uhrzeit des Dateneingangs, ein ggf. vergebener Eingangstempel und die den Dateneingang verarbeitende Anwendung dokumentiert werden, sowie die Form des Dateneingangs, nämlich als Papierformular (wenn ein Belegleser das DatML/RES-Dokument erzeugt hat), als Online-Formular oder als Datei.

Beispiel Online-Erhebung (HTML-Formular):

```

<dateneingang>
  <datum>20020506</datum>
  <uhrzeit>195412</uhrzeit>
  <eingangsstempel>w320020506195412-230659</eingangsstempel>
  <anwendung>
    <anwendungsname>IDEV</anwendungsname>
    <version>1.0</version>
  </anwendung>
  <onlineformular>
    <formularname>FB-23111-2003</formularname>
  </onlineformular>
</dateneingang>
  
```

```
<formularID>20020506195412</formularID>
</onlineformular>
</dateneingang>
```

Beispiel Dateilieferung; die Datei *intra200204*, erstellt am *12. Mai 2002* um *11:34:02* Uhr, wurde auf Diskette *437161* mit Datensätzen im *CSV-Format* geliefert und am *15. Mai 2002* um *14:19:21* Uhr von der Anwendung *DTEReg* gelesen und nach *DatML/RAW* konvertiert:

```
<dateneingang>
  <datum>20020515</datum>
  <uhrzeit>141921</uhrzeit>
  <eingangsstempel>w20020515141921-863275</eingangsstempel>
  <anwendung>
    <anwendungsname>DTEReg</anwendungsname>
    <version>1.5.1</version>
  </anwendung>
  <datei>
    <dateiname>intra200204</dateiname>
    <datum>20020512</datum>
    <uhrzeit>113402</uhrzeit>
    <datenformat typ="csv"/>
    <datentraeger typ="diskette">437161</datentraeger>
  </datei>
</dateneingang>
```

### 4.5.3. Element <dateiname>

<b>Element</b>	<dateiname>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de

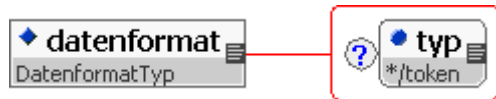


#### Details

Der Name einer Datei.

#### 4.5.4. Element <datenformat>

<b>Element</b>	<datenformat>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Attribute: **S Datentyp, Default, Werte**

typ	<b>D enumeration:</b>
	<ul style="list-style-type: none"> <li>• <u>undefiniert</u></li> <li>• datml</li> <li>• xml</li> <li>• datensatz</li> <li>• edifact</li> </ul>

Typ des Datenformates; der Wert `datml` zeigt an, dass eine DatML/RES-Instanz geliefert wurde.

#### Details

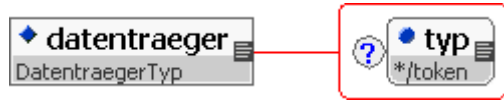
<datenformat> beschreibt das für die Rohdatenlieferung verwendete Datenformat. Es können ein Typ (gewöhnliche Datensätze, XML oder EDIFACT) sowie der Name des Datenformates angegeben werden.

#### Beispiel:

```
<datenformat typ="xml">INSTAT/XML</datenformat>
```

#### 4.5.5. Element <datentraeger>

<b>Element</b>	<datentraeger>
Datentyp:	<i>empty-element</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



<b>Attribute:</b>	<b>S</b> <b>Datentyp, Default, Werte</b>
typ	D <i>enumeration</i> : <u>undefiniert</u>   diskette   cdrom   dvd   kassette   magnetband Typ des Datenträgers

#### Details

<datentraeger> gibt den Namen und Typ des verwendeten Datenträgers an.

Beispiel:

```
<datentraeger typ="diskette">STBA 330827</datenformat>
```

#### 4.5.6. Element <dokumentID>

<b>Element</b>	<dokumentID>
Datentyp:	<i>empty-element</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



<b>Attribute:</b>	<b>S</b> <b>Datentyp, Default, Werte</b>
klasse	D <i>char.default</i> ; Optionale Klasse oder Klassifikation

#### Details

<dokumentID> enthält eine Angabe für die (eindeutige) Identifizierung einer XML-Dokumentinstanz (zum Beispiel einer Instanz von DatML/RES oder DatML/RAW).

#### 4.5.7. Element <eingangsstempel>

<b>Element</b>	<eingangsstempel>
Datentyp:	<i>empty-element</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



<b>Attribute:</b>	<b>S</b> <b>Datentyp, Default, Werte</b>
klasse	D <i>char.default</i> ; Optionale Klasse oder Klassifikation

#### Details

<eingangsstempel> dient zur Ablage eines beim Dateneingang erzeugten Eingangsstempels.

#### 4.5.8. Element <formularID>

<b>Element</b>	<formularID>
Datentyp:	<i>empty-element</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



<b>Attribute:</b>	<b>S</b> <b>Datentyp, Default, Werte</b>
klasse	D <i>char.default</i> ; Optionale Klasse oder Klassifikation

#### Details

<formularID> enthält eine Angabe für die Identifizierung eines einzelnen oder einer zusammenhängenden Menge von Online- bzw. Papierformularen, etwa für einen Berichtszeitraum einer Erhebung. <formularID> kann z.B. mit den Barcode eines Papierformulars enthalten.

#### 4.5.9. Element <formulardname>

<b>Element</b>	<formulardname>
Datentyp:	<i>empty-element</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



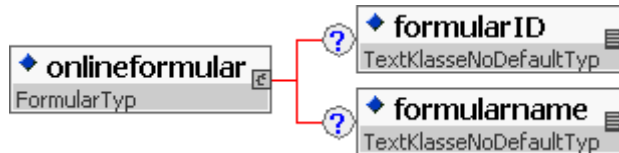
<b>Attribute:</b>	<b>S</b> <b>Datentyp, Default, Werte</b>
klasse	D <i>char.default</i> ; Optionale Klasse oder Klassifikation

#### Details

<formulardname> enthält den Namen eines Online- bzw. Papierformulars. Für die Identifizierung eines einzelnen oder einer Serie von Formularen soll das Element <formulardID> verwendet werden.

#### 4.5.10. Element `<onlineformular>`

<b>Element</b>	<code>&lt;onlineformular&gt;</code>
Datentyp:	<i>element-content</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>

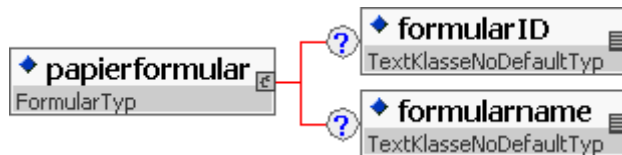


#### Details

`<onlineformular>` zeigt an, dass die Rohdaten mit einem Online-Formular (z.B. HTML) geliefert wurden. Hat das Formular einen Namen, kann dieser mit dem Element `<formularname>` angegeben werden. Für die Identifizierung eines einzelnen Beleges oder einer Serie von Formularen (z.B. für einen Berichtszeitraum einer Erhebung) steht das Element `<formularID>` zur Verfügung.

#### 4.5.11. Element `<papierformular>`

<b>Element</b>	<code>&lt;papierformular&gt;</code>
Datentyp:	<i>element-content</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

`<papierformular>` zeigt an, dass die Rohdaten auf einem Papierformular geliefert wurden. Hat das Formular einen Namen, kann dieser mit dem Element `<formularname>` angegeben werden. Für die Identifizierung eines einzelnen Formulars oder einer Serie von Formularen (z.B. für einen Berichtszeitraum einer Erhebung) steht das Element `<formularID>` zur Verfügung.

#### 4.5.12. Element <upload>

<b>Element</b>	<upload>
Datentyp:	<i>empty-element</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



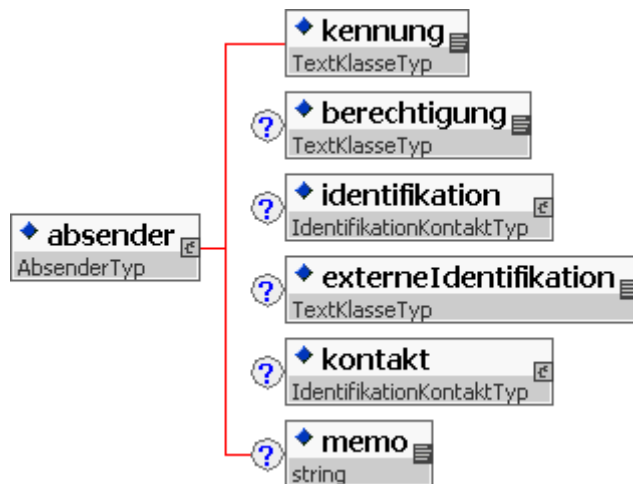
#### Details

<upload> zeigt an, dass eine Datei per Upload geliefert wurde. Eine gesonderte Protokollierung von Datum und Uhrzeit des *absenderseitigen Uploads* ist nicht vorgesehen. Stattdessen sollen im Element <dateneingang> Datum und Uhrzeit des *empfängerseitigen Dateieingangs* dokumentiert werden.

## 4.6. Verfahrensteilnehmer, Identifikation, Kontaktangaben

### 4.6.1. Element <absender>

<b>Element</b>	<absender>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de

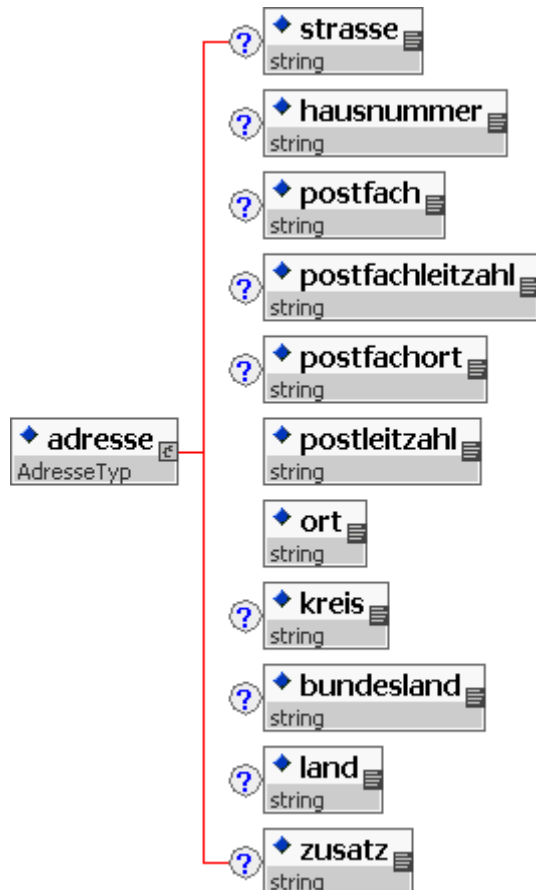


#### Details

Dieses Element beschreibt den Absender eines DatML/RES-Dokumentes. Es ist anwendungsspezifisch festzuliegen, welche Daten zu Verifikation der Identität des Absenders herangezogen werden.

#### 4.6.2. Element <adresse>

<b>Element</b>	<adresse>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

<adresse> bezeichnet eine postalische Adresse, erweitert um die Möglichkeit, Kreis und Bundesland anzugeben.

### 4.6.3. Element <anrede>

<b>Element</b>	<anrede>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>

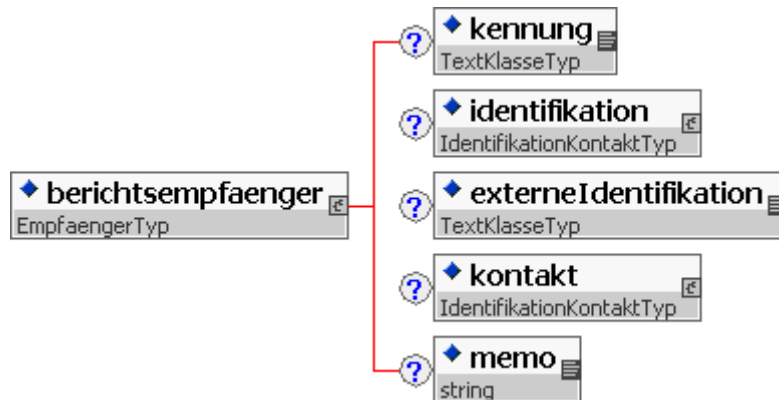


#### Details

Die Anrede einer Person.

#### 4.6.4. Element <berichtsempfaenger>

<b>Element</b>	<berichtsempfaenger>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de

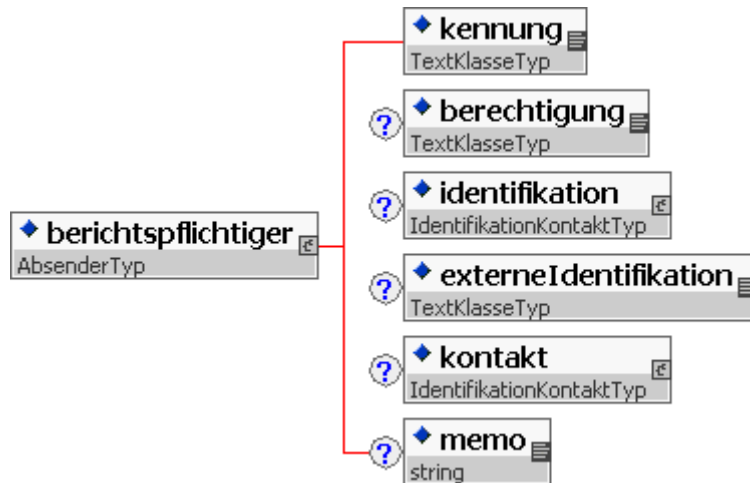


#### Details

Der Berichtsempfänger ist die erhebende Stelle, i.d.R. ein Statistisches Landesamt oder das Statistische Bundesamt.

#### 4.6.5. Element <berichtspflichtiger>

<b>Element</b>	<berichtspflichtiger>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

Der Berichtspflichtige, in dessen Namen die Meldung erfolgt.

#### 4.6.6. Element <bundesland>

<b>Element</b>	<bundesland>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

Der Name eines Bundeslandes oder einer anderen entsprechenden Gebiets- oder Verwaltungseinheit.

#### 4.6.7. Element <email>

<b>Element</b>	<email>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>

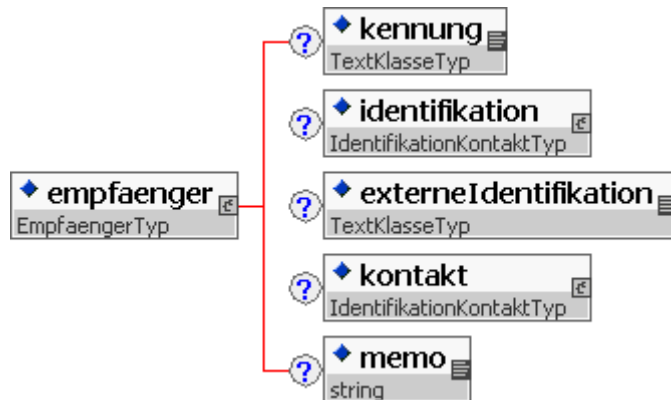


#### Details

Eine E-Mail-Adresse.

#### 4.6.8. Element <empfaenger>

<b>Element</b>	<empfaenger>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

Der Empfänger eines Dokumentes muss in der Lage sein, sich selbst als Adressaten zu bestätigen und benötigt deshalb eine Angabe zu seiner Identifikation.

#### 4.6.9. Element <externeIdentifikation>

<b>Element</b>	<externeIdentifikation>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



<b>Attribute:</b>	<b>S</b> <b>Datentyp, <u>Default</u>, Werte</b>
klasse	? <i>char</i> ; Optionale Klasse oder Klassifikation

#### Details

Eine externe (nicht von den statistischen Ämtern festgelegte) Angabe für die Identifikation eines Verfahrensbeteiligten.

#### 4.6.10. Element `<fax>`

<b>Element</b>	<code>&lt;fax&gt;</code>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

Eine Fax-Nummer.

#### 4.6.11. Element `<hausnummer>`

<b>Element</b>	<code>&lt;hausnummer&gt;</code>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>

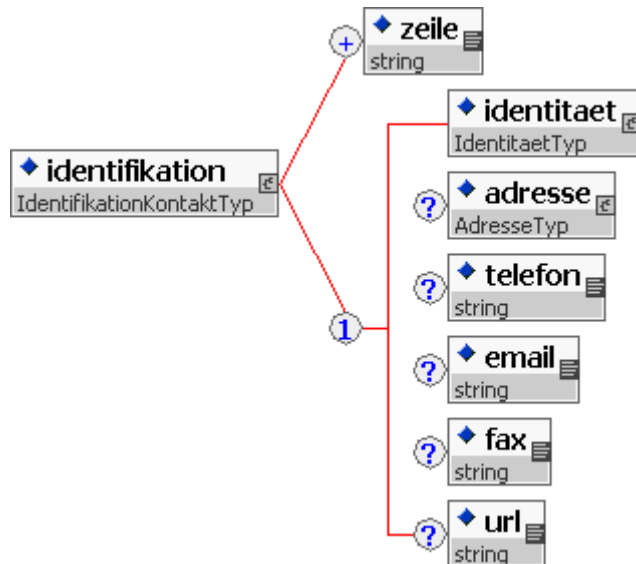


#### Details

Eine Hausnummer.

#### 4.6.12. Element <identifikation>

<b>Element</b>	<identifikation>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de

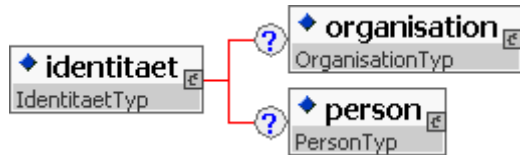


#### Details

<identifikation> enthält die Angaben zur Beschreibung der natürlichen oder juristischen Person eines Verfahrensteilnehmers und erlaubt wahlweise zeilenbasierte oder strukturierte Angaben.

#### 4.6.13. Element <identitaet>

<b>Element</b>	<identitaet>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de

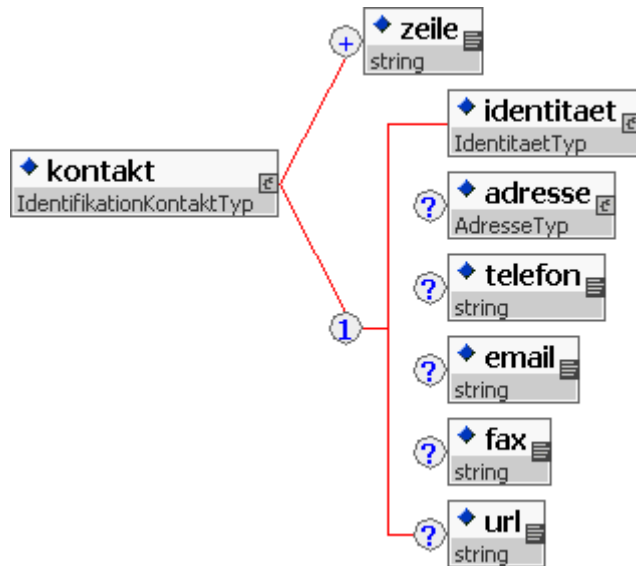


#### Details

Die Identität eines Verfahrensteilnehmers ist die einer Person und/oder Organisation.

#### 4.6.14. Element <kontakt>

<b>Element</b>	<kontakt>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

<kontakt> erlaubt wahlweise zeilenbasierte oder strukturierte Angaben.

#### 4.6.15. Element <kreis>

<b>Element</b>	<kreis>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

Der Name eines Kreises oder einer anderen entsprechenden Gebiets- oder Verwaltungseinheit; in anderen Staaten könnte dies z.B. einem Bezirk, Distrikt oder Departement entsprechen.

#### 4.6.16. Element <land>

<b>Element</b>	<land>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

Der Name eines Nationalstaates oder einer anderen entsprechenden Gebiets- oder Verwaltungseinheit.

#### 4.6.17. Element <nachname>

<b>Element</b>	<nachname>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

Der oder die Nachnamen einer natürlichen Person.

#### 4.6.18. Element <niederlassung>

<b>Element</b>	<niederlassung>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de

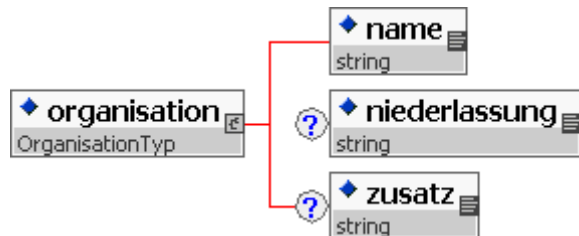


#### Details

Der Name der Niederlassung oder einer anderen nachgeordneten Einheit einer Organisation.

#### 4.6.19. Element `<organisation>`

<b>Element</b>	<code>&lt;organisation&gt;</code>
Datentyp:	<i>element-content</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

`<organisation>` beschreibt eine juristische, wirtschaftliche, politische oder sonstige Organisationseinheit, etwa eine Firma, eine Behörde oder eine internationale Institution (im Ggs. zu einer natürlichen Person).

#### 4.6.20. Element <ort>

<b>Element</b>	<ort>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>

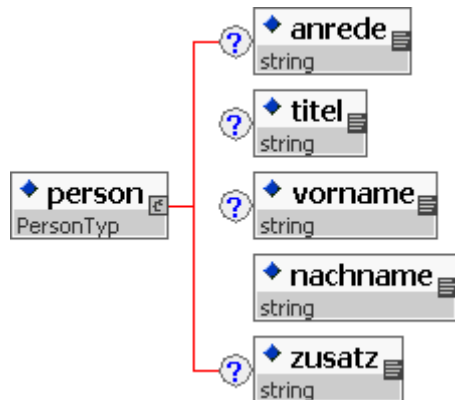


#### Details

Der Name einer Stadt, einer Gemeinde oder einer anderen entsprechenden Gebiets- oder Verwaltungseinheit.

#### 4.6.21. Element <person>

<b>Element</b>	<person>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

Eine Person, die mindestens durch ihre(n) Nachnamen identifiziert wird.

#### 4.6.22. Element `<postfach>`

<b>Element</b>	<code>&lt;postfach&gt;</code>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

Eine Postfachangabe.

#### 4.6.23. Element `<postfachleitzahl>`

<b>Element</b>	<code>&lt;postfachleitzahl&gt;</code>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

Die Leitzahl eines Postfachs.

#### 4.6.24. Element <postfachort>

<b>Element</b>	<postfachort>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

Der Ort eines Postfachs.

#### 4.6.25. Element `<postleitzahl>`

<b>Element</b>	<code>&lt;postleitzahl&gt;</code>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

Eine Postleitzahl.

#### 4.6.26. Element <strasse>

<b>Element</b>	<strasse>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

Der Name einer Straße oder eine entsprechende Adressangabe unterhalb der Ebene des Ortes.

#### 4.6.27. Element <telefon>

<b>Element</b>	<telefon>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

Eine beliebige Telefonnummer.

#### 4.6.28. Element <titel>

<b>Element</b>	<titel>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

Adelstitel, akademische Titel oder sonstige Titel einer natürlichen Person.

#### 4.6.29. Element `<url>`

<b>Element</b>	<code>&lt;url&gt;</code>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

Ein URL (Universal Resource Locator), z.B. einer Internetseite.

#### 4.6.30. Element <vorname>

<b>Element</b>	<vorname>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

Der oder die Vornamen einer natürlichen Person.

#### 4.6.31. Element <zeile>

<b>Element</b>	<zeile>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



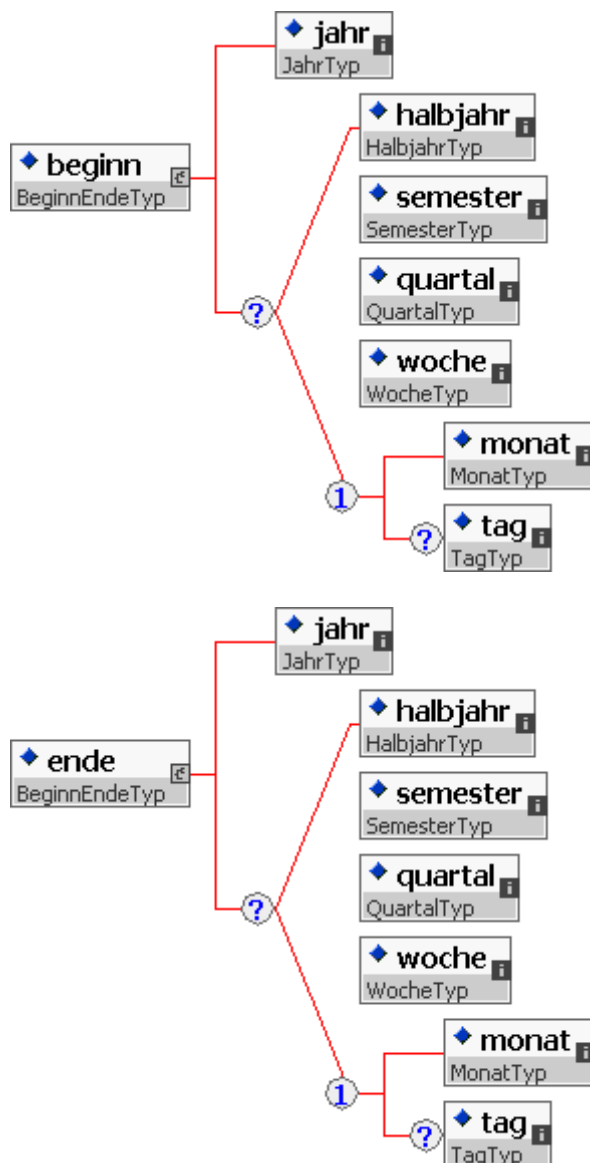
#### Details

Eine Zeile eines zeilenbasierten Datenmodells (s. 4.6.12).

## 4.7. Erhebung, Berichtszeitraum

### 4.7.1. Elemente <beginn>, <ende>

<b>Element</b>	<beginn>, <ende>
Datentyp:	<i>element-content</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>

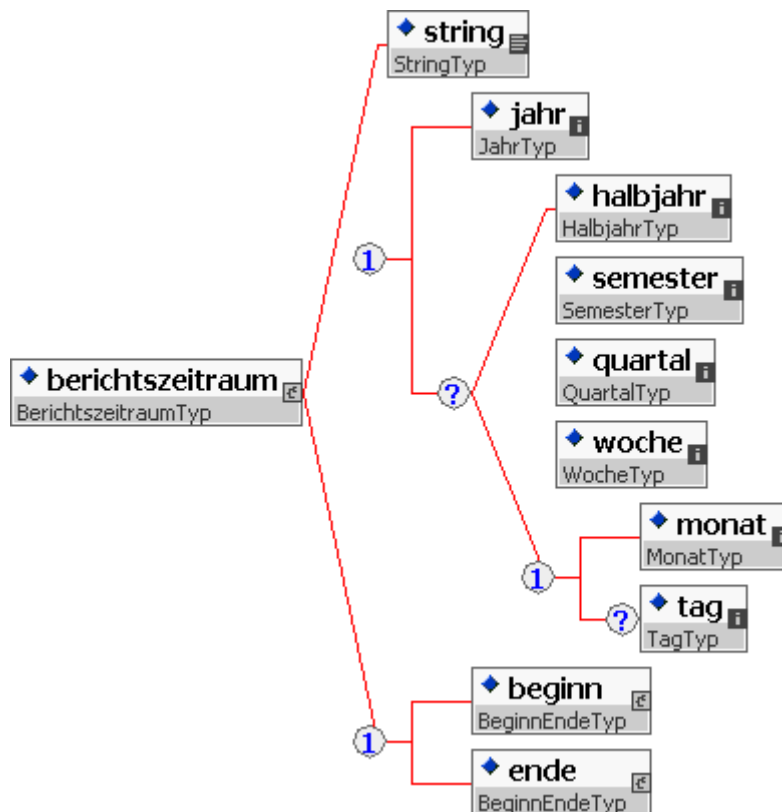


#### Details

Beginn oder Ende eines Zeitraumes in Form eines Kalender-, Halbjahres-, Vierteljahres-, Wochendatums oder Semesters.

#### 4.7.2. Element <berichtszeitraum>

<b>Element</b>	<berichtszeitraum>
<b>Datentyp:</b>	<i>element-content</i>
<b>Namespace:</b>	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

<berichtszeitraum> gibt den Berichtszeitraum an; dies ist ein Einzeldatum, beginnend mit der obligatorischen Angabe des Jahres, oder eine Spanne aus zwei Einzeldaten, die Beginn und Ende des Berichtszeitraumes markieren. Die Angaben sind einschließliche Werte. In Spannen sind verschiedene Zeiträume für Beginn und Ende möglich, z.B. „zweites Halbjahr 1999 bis einschließlich drittes Quartal 2000“.

Ein Tag kann nur in Verbindung mit einem Monat angegeben werden.

Beispiel:

```
<berichtszeitraum>
  <beginn>
    <jahr>1999</jahr><halbjahr>2</halbjahr>
  </beginn>
  <ende>
    <jahr>2000</jahr><quartal>3</quartal>
  </ende>
</berichtszeitraum>
```

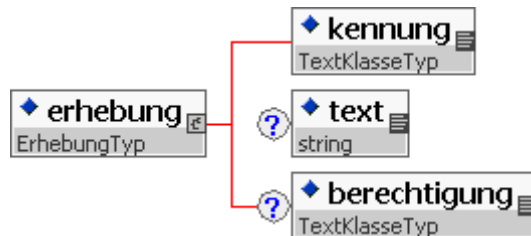
Der Berichtszeitraum kann auch eine formatierte Zeichenkette sein:

```
<berichtszeitraum>  
  <string klasse="datum" format="jjjjh-jjjjq">1999H2-2000Q3</string>  
</berichtszeitraum>
```

(siehe Element `<string>`). Die Festlegung zulässiger Datumsangaben und weitere Regeln (invertierte Spannen,..) sind Sache der Anwendung.

### 4.7.3. Element <erhebung>

<b>Element</b>	<erhebung>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

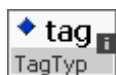
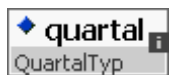
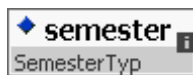
Eine Erhebungsangabe hat eine Kennung (<kennung>) und einen optionalen Namen (<name>). Stammt die Kennung aus einer Klassifikation oder Systematik, kann diese mit dem Attribut `klasse` des Elementes <kennung> angegeben werden. Es ist außerdem möglich, eine Authentifizierungsangabe (<berechtigung>) abzulegen, da diese erhebungsabhängig sein kann.

#### 4.7.4. Elemente <jahr>, <halbjahr>, <semester>, <quartal>, <monat>, <woche>, <tag>

**Element** <jahr>, <halbjahr>, <semester>, <quartal>, <monat>, <woche>, <tag>

Datentyp: *char*

Namespace: <http://www.destatis.de/schema/datml-res/1.0/de>



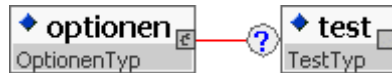
#### Details

Die Elemente sind Bestandteile von Datumsangaben. Der Wertebereich der Elemente entspricht denen in formatierten Datumsangaben (2.6.7).

## 4.8. Verarbeitungssteuerung

### 4.8.1. Element <optionen>

<b>Element</b>	<optionen>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

Benutzeroptionen, s 2.6.4.

## 4.8.2. Element <test>

<b>Element</b>	<test>
Datentyp:	<i>empty-element</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



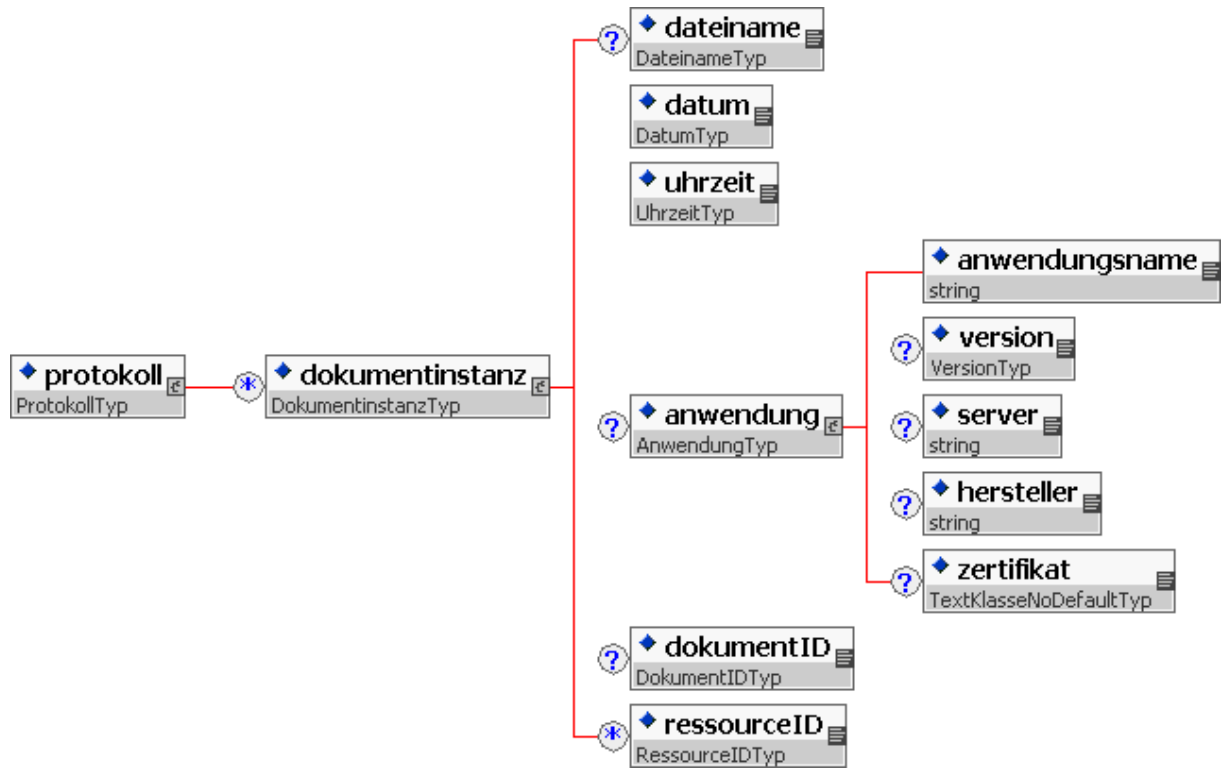
<b>Attribute:</b>	<b>S</b> <b>Datentyp, Default, Werte</b>
kennung	? <i>char.</i> ; Eine optionale Testkennung

### Details

<test> zeigt an, dass die Dokumentinstanz nur zu Testzwecken versendet wurde. Das Attribut `kennung` erlaubt eine beliebige Angabe, z.B. für die Steuerung und Kontrolle von Testserien.

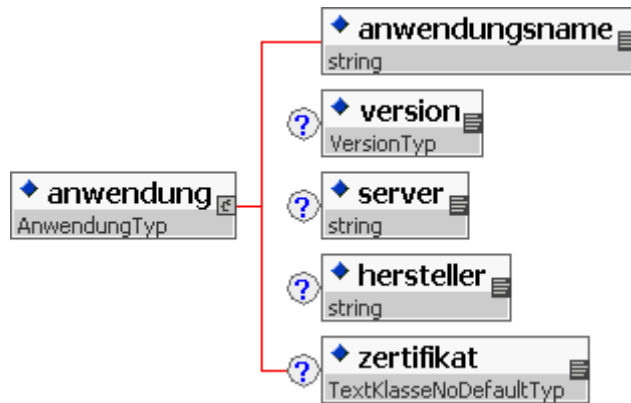
## 4.9. Protokollierung

### 4.9.1. Übersicht



#### 4.9.2. Element <anwendung>

<b>Element</b>	<anwendung>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

<anwendung> beschreibt eine Anwendung; das Format der Versionsbezeichnung bzw. des Zertifizierungskennzeichens ist anwendungsabhängig..

### 4.9.3. Element <anwendungsname>

**Element** <anwendungsname>

Datentyp: *char*

Namespace: <http://www.destatis.de/schema/datml-res/1.0/de>

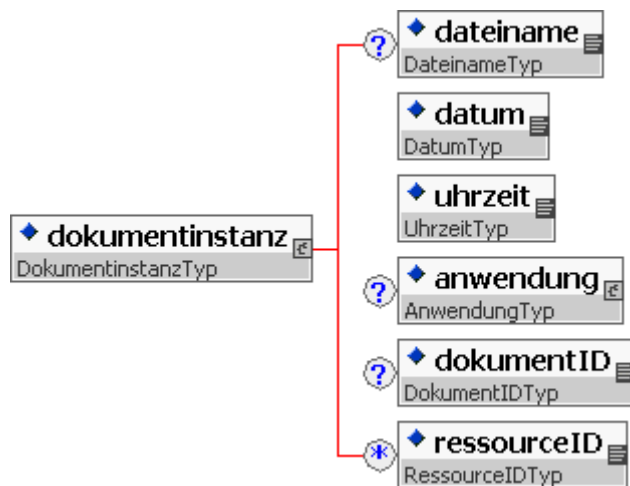


#### Details

Der Name einer (Software-)Anwendung.

#### 4.9.4. Element <dokumentinstanz>

<b>Element</b>	<dokumentinstanz>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

<dokumentinstanz> erlaubt die Dokumentierung von DatML/RES-Dokumentinstanzen. Es ist möglich, eine Dokument-ID anzugeben und die erzeugte Datei sowie die erzeugende Anwendung zu beschreiben.

Die dokumenterzeugende Anwendung kann die von ihr verwendeten Ressourcen, wie etwa ein DatML/SDF-Dokument mit einer Erhebungsbeschreibung, mit Hilfe des Elementes <ressourceID> dokumentieren.

#### 4.9.5. Element <hersteller>

<b>Element</b>	<hersteller>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

Die Bezeichnung des Herstellers einer Anwendung.

#### 4.9.6. Element <protokoll>

<b>Element</b>	<protokoll>
Datentyp:	<i>element-content</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de

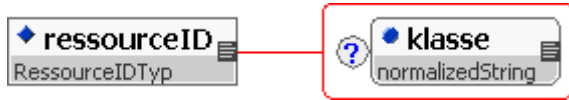


#### Details

<protokoll> erlaubt die die Dokumentation beliebig vieler DatML/RES-Dokumentinstanzen (<dokumentinstanz>).

#### 4.9.7. Element <ressourceID>

<b>Element</b>	<ressourceID>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



<b>Attribute:</b>	<b>S</b> <b>Datentyp, <u>Default</u>, Werte</b>
klasse	D <i>char</i> : <u>default</u> ; Optionale Klasse oder Klassifikation

#### Details

Die Kennzeichnung einer für die Dokumentgenerierung verwendeten Ressource.

#### 4.9.8. Element <server>

<b>Element</b>	<server>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

Die Bezeichnung eines Servers.

#### 4.9.9. Element <zertifikat>

<b>Element</b>	<zertifikat
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



<b>Attribute:</b>	<b>S</b> <b>Datentyp, <u>Default</u>, Werte</b>
klasse	D <i>char</i> . <u>default</u> ; Optionale Klasse oder Klassifikation

#### Details

Ein Zertifikatsschlüssel. Anwendungen können die Angabe eines Zertifikatsschlüssels vorschreiben.

## 4.10. Sonstige Elemente

### 4.10.1. Element <berechtigung>

<b>Element</b>	<berechtigung>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



<b>Attribute:</b>	<b>S</b> <b>Datentyp, <u>Default</u>, Werte</b>
klasse	D <i>char:<u>default</u></i> ; Optionale Klasse oder Klassifikation

#### Details

Das Element <berechtigung> soll für Angaben verwendet werden, die ein Erhebungsverfahren ggf. über die <kennung> (und Namens- und Adressangaben) hinaus als zusätzlichen Nachweis für die Zulassung des Anwenders verlangt (z.B. ein Zugangsschlüssel o.ä.).

#### 4.10.2. Element <datum>

<b>Element</b>	<datum>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



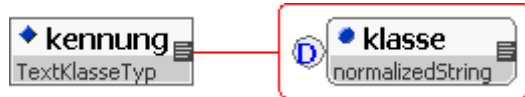
<b>Attribute:</b>	<b>S</b> <b>Datentyp, <u>Default</u>, Werte</b>
format	F <i>char</i> : <u>jjjjmmtt</u> ; Formatschema
klasse	F <i>char</i> : <u>datum</u> ; Klasse des Formatschemas

#### Details

Ein Datum in Form eines Strings; Formatschema und Formatklasse sind fest vorgegeben (2.6.7).

### 4.10.3. Element <kennung>

<b>Element</b>	<kennung>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



<b>Attribute:</b>	<b>S</b> <b>Datentyp, <u>Default</u>, Werte</b>
klasse	D <i>char</i> : <u>default</u> ; Optionale Klasse oder Klassifikation

#### Details

Eine Kennungsangabe je nach Kontext, z.B. User-ID, Materialnummer...

#### 4.10.4. Element `<memo>`

<b>Element</b>	<code>&lt;memo&gt;</code>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

`<memo>` ist an vielen Stellen zur Ablage eines Kommentars zulässig.

#### 4.10.5. Element `<name>`

<b>Element</b>	<code>&lt;name&gt;</code>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

Ein beliebiger Name, dessen Bedeutung kontextabhängig ist:

```
<organisation><name>Milch AG</name></organisation>
```

#### 4.10.6. Element `<string>`

<b>Element</b>	<code>&lt;string&gt;</code>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



<b>Attribute:</b>	<b>S</b> <b>Datentyp, <u>Default</u>, Werte</b>
<code>format</code>	M <i>char</i> ; Formatstring (Template).
<code>klasse</code>	M <i>enumeration</i> : <u>undefiniert</u>   datum; Formatklasse.

#### Details

`<string>` ermöglicht die Angabe formatierter Zeichenketten. Derzeitig ist nur die Verwendung von Datumsangaben definiert (s. 2.6.7).

```
<string klasse="datum" format="jjmmtt">020815</string>
```

#### 4.10.7. Element <text>

<b>Element</b>	<text>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

Ein beliebiger Text mit kontextabhängiger Bedeutung.

#### 4.10.8. Element <uhrzeit>

<b>Element</b>	<uhrzeit>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



<b>Attribute:</b>	<b>S</b> <b>Datentyp, <u>Default</u>, Werte</b>
format	F <i>char</i> : <u>hhmmss</u> ; Formatschema
klasse	F <i>char</i> : <u>datum</u> ; Klasse des Formatschemas

#### Details

Eine Uhrzeit in Form eines Strings; Formatschema und Formatklasse sind fest vorgegeben (2.6.7). Wird die Uhrzeit nicht sekunden- oder sogar minutengenau angegeben, sind die entsprechenden Felder mit Nullen zu belegen.

#### 4.10.9. Element `<version>`

<b>Element</b>	<code>&lt;version&gt;</code>
Datentyp:	<i>char</i>
Namespace:	<a href="http://www.destatis.de/schema/datml-res/1.0/de">http://www.destatis.de/schema/datml-res/1.0/de</a>



#### Details

Eine Versionsangabe mit kontextabhängiger Bedeutung.

#### 4.10.10. Element <zusatz>

<b>Element</b>	<zusatz>
Datentyp:	<i>char</i>
Namespace:	http://www.destatis.de/schema/datml-res/1.0/de



#### Details

Ein Zusatz mit kontextabhängiger Bedeutung als generische Erweiterung eines Datenmodells.

## 5. Beispieldokument *[nicht-normativ]*

Das Beispieldokument enthält eine Meldung für die Gewerbeanzeigestatistik mit fiktiven Daten.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<DatML-RES-D version="1.0" xmlns="http://www.destatis.de/schema/datml-res/1.0/de">
  <absender>
    <kennung klasse="STAIID">04</kennung>
  </absender>
  <empfaenger>
    <kennung klasse="MELDID">HB492949</kennung>
  </empfaenger>
  <pruefprotokoll>
    <pruefung>
      <pruefstufe>3</pruefstufe>
      <dokumenttyp>
        <name>Datml/RAW</name>
        <version>datml-raw-de-2_0.xsd</version>
      </dokumenttyp>
      <syntax pruefstatus="fehlerfrei"/>
      <semantik pruefstatus="fehlerfrei"/>
      <autorisierung pruefstatus="fehlerfrei"/>
      <daten pruefstatus="fehlerhaft"/>
    </pruefung>
    <dokument pruefstatus="fehlerhaft" dokumentstatus="akzeptiert">
      <dateneingang>
        <datum>20041031</datum>
        <uhrzeit>152609</uhrzeit>
        <eingangsstempel/>
        <anwendung>
          <anwendungsname>IDEV 0.5 (Beta)</anwendungsname>
        </anwendung>
        <onlineformular>
          <formularname>FB-GAW-2005</formularname>
        </onlineformular>
      </dateneingang>
    </dokument>
  </pruefprotokoll>
  <nachricht pruefstatus="fehlerhaft">
    <meldung pruefstatus="fehlerhaft">
      <meldungsID>40Q78H4U342S094K60M421315309X3S4</meldungsID>
      <meldungsart>
        <original/>
      </meldungsart>
      <erhebung>
        <kennung klasse="ERHID">12345</kennung>
      </erhebung>
      <berichtszeitraum>
        <jahr>2004</jahr>
      </berichtszeitraum>
      <berichtspflichtiger>
        <kennung klasse="MELDID">987654321</kennung>
      </berichtspflichtiger>
      <berichtsempfaenger>
        <kennung klasse="STAIID">99</kennung>
      </berichtsempfaenger>
      <fehler>
        <schluessel>11201</schluessel>
        <gewicht>1</gewicht>
        <text>Der Wert ist zu groß</text>
        <position format="xpath">/DatML-RAW-
D/nachricht[1]/segment[2]/datensegment[1]/satz[269]/mm[8]</position>
        <merkmal>Alter</merkmal>
      </fehler>
    </meldung>
  </nachricht>
</DatML-RES-D>
```

```
</nachricht>  
</pruefprotokoll>  
</DatML-RES-D>
```

## 6. XML-Schema 1.0 [nicht-normativ]

Dieser Ausdruck der XML-Schema-Definition (XSD) kann nicht als Referenz verwendet werden.

```
<?xml version = "1.0" encoding = "ISO-8859-1"?>
<xs:schema xmlns = "http://www.destatis.de/schema/datml-res/1.0/de"
  targetNamespace = "http://www.destatis.de/schema/datml-res/1.0/de"
  xmlns:xs = "http://www.w3.org/2001/XMLSchema"
  version = "1.0"
  elementFormDefault = "qualified">
  <xs:annotation>
    <xs:documentation xml:lang = "de" source = "http://www.statspez.de">
      Schema      : DatML/RES
      Version     : 1.0
      Variante    : D
      Edition     : 0
      Typ        : XML-Schema
      Namensraum: http://www.destatis.de/schema/datml-res/1.0/de
      Stand      : 20041209
      Autor      : Michael Schaefer
                  Statistisches Bundesamt
                  Gustav-Stresemann-Ring 11
                  D-65189 Wiesbaden
                  Tel. +49(0)611/75-3652
                  Fax +49(0)611/72-4000
                  mailto:michael.schaefer@destatis.de

      Beschreibung:

      Updates (U) und Korrekturen (K):

      *** Version 1.0.0 ***

      001 K YYYYMMDD/K/1.0.0/D//VORNAME_NAME_AUTOR

    </xs:documentation>
  </xs:annotation>

  <!-- Allgemeine Typen -->

  <xs:complexType name = "DatumTyp">
    <xs:simpleContent>
      <xs:extension base = "xs:string">
        <xs:attribute name = "format" fixed = "jjjjmmtt" type = "xs:normalizedString"/>
        <xs:attribute name = "klasse" fixed = "datum" type = "xs:normalizedString"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>

  <xs:complexType name = "TextKlasseTyp">
    <xs:simpleContent>
      <xs:extension base = "xs:string">
        <xs:attribute name = "klasse" default = "default" type = "xs:normalizedString"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>

  <xs:complexType name = "TextKlasseNoDefaultTyp">
    <xs:simpleContent>
      <xs:extension base = "xs:string">
        <xs:attribute name = "klasse" type = "xs:normalizedString"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>

  <xs:complexType name = "UhrzeitTyp">
    <xs:simpleContent>
      <xs:extension base = "xs:string">
        <xs:attribute name = "format" fixed = "hhmmss" type = "xs:normalizedString"/>
        <xs:attribute name = "klasse" fixed = "datum" type = "xs:normalizedString"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
```

```

    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name = "VersionTyp">
  <xs:simpleContent>
    <xs:extension base = "xs:string"/>
  </xs:simpleContent>
</xs:complexType>

<!-- Typen Optionen -->

<xs:complexType name="OptionenTyp">
  <xs:sequence>
    <xs:element name = "test" type="TestTyp" minOccurs = "0"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="TestTyp">
  <xs:attribute name = "kennung" type = "xs:normalizedString"/>
</xs:complexType>

<!-- Typen Protokollierung -->

<xs:complexType name = "AnwendungTyp">
  <xs:sequence>
    <xs:element name = "anwendungsname" type = "xs:string"/>
    <xs:element name = "version" type = "VersionTyp" minOccurs = "0"/>
    <xs:element name = "server" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "hersteller" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "zertifikat" type = "TextKlasseNoDefaultTyp" minOccurs = "0"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name = "DateneingangTyp">
  <xs:sequence>
    <xs:element name = "datum" type = "DatumTyp"/>
    <xs:element name = "uhrzeit" type = "UhrzeitTyp"/>
    <xs:element name = "eingangsstempel" type = "TextKlasseNoDefaultTyp"/>
    <xs:element ref = "anwendung"/>
    <xs:choice minOccurs = "0">
      <xs:element name = "papierformular" type = "FormularTyp"/>
      <xs:element name = "onlineformular" type = "FormularTyp"/>
      <xs:element name = "datei" type = "DateiTyp"/>
    </xs:choice>
  </xs:sequence>
</xs:complexType>

<xs:complexType name = "DokumentinstanzTyp">
  <xs:sequence>
    <xs:element name = "dateiname" type = "DateinameTyp" minOccurs = "0"/>
    <xs:element name = "datum" type = "DatumTyp"/>
    <xs:element name = "uhrzeit" type = "UhrzeitTyp"/>
    <xs:element ref = "anwendung" minOccurs = "0"/>
    <xs:element name = "dokumentID" type = "DokumentIDTyp" minOccurs = "0"/>
    <xs:element name = "ressourceID" type = "RessourceIDTyp" minOccurs = "0" maxOc-
curs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name = "DateiTyp">
  <xs:sequence>
    <xs:element name = "dateiname" type = "DateinameTyp" minOccurs = "0"/>
    <xs:element name = "datum" type = "DatumTyp" minOccurs = "0"/>
    <xs:element name = "uhrzeit" type = "UhrzeitTyp" minOccurs = "0"/>
    <xs:element ref = "anwendung" minOccurs = "0"/>
    <xs:element name = "dokumentID" type = "DokumentIDTyp" minOccurs = "0"/>
    <xs:element name = "datenformat" type = "DatenformatTyp" minOccurs = "0"/>
    <xs:choice minOccurs = "0">
      <xs:element name = "datentraeger" type = "DatentraegerTyp"/>
      <xs:element name = "upload"/>
    </xs:choice>
  </xs:sequence>
</xs:complexType>

<xs:complexType name = "DateinameTyp">

```

```

    <xs:simpleContent>
      <xs:extension base = "xs:string"/>
    </xs:simpleContent>
  </xs:complexType>

  <xs:complexType name = "DatenformatTyp">
    <xs:simpleContent>
      <xs:extension base = "xs:string">
        <xs:attribute name = "typ">
          <xs:simpleType>
            <xs:restriction base = "xs:token">
              <xs:enumeration value = "undefiniert"/>
              <xs:enumeration value = "datml"/>
              <xs:enumeration value = "xml"/>
              <xs:enumeration value = "datensatz"/>
              <xs:enumeration value = "csv"/>
              <xs:enumeration value = "edifact"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>

  <xs:complexType name = "DatentraegerTyp">
    <xs:simpleContent>
      <xs:extension base = "xs:string">
        <xs:attribute name = "typ">
          <xs:simpleType>
            <xs:restriction base = "xs:token">
              <xs:enumeration value = "undefiniert"/>
              <xs:enumeration value = "diskette"/>
              <xs:enumeration value = "cdrom"/>
              <xs:enumeration value = "dvd"/>
              <xs:enumeration value = "kassette"/>
              <xs:enumeration value = "magnetband"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>

  <xs:complexType name = "FormularTyp">
    <xs:sequence>
      <xs:element name = "formularID" type = "TextKlasseNoDefaultTyp" minOccurs = "0"/>
      <xs:element name = "formularname" type = "TextKlasseNoDefaultTyp" minOccurs = "0"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name = "DokumentIDTyp">
    <xs:simpleContent>
      <xs:extension base = "TextKlasseNoDefaultTyp"/>
    </xs:simpleContent>
  </xs:complexType>

  <xs:complexType name = "RessourceIDTyp">
    <xs:simpleContent>
      <xs:extension base = "TextKlasseNoDefaultTyp"/>
    </xs:simpleContent>
  </xs:complexType>

  <xs:complexType name = "ProtokollTyp">
    <xs:sequence>
      <xs:element name = "dokumentinstanz" type = "DokumentinstanzTyp" minOccurs = "0" maxOccurs = "unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <!-- Typen der Identifikation (Beteiligte) -->

  <xs:complexType name = "IdentifikationKontaktTyp">
    <xs:choice>
      <xs:element name = "zeile" type = "xs:string" maxOccurs = "unbounded"/>
      <xs:sequence>
        <xs:element name = "identitaet" type = "IdentitaetTyp"/>
      </xs:sequence>
    </xs:choice>
  </xs:complexType>

```

```

    <xs:element name = "adresse" type = "AdresseTyp" minOccurs = "0"/>
    <xs:element name = "telefon" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "email" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "fax" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "url" type = "xs:string" minOccurs = "0"/>
  </xs:sequence>
</xs:choice>
</xs:complexType>

<xs:complexType name = "IdentitaetTyp">
  <xs:sequence>
    <xs:element name = "organisation" type = "OrganisationTyp" minOccurs = "0"/>
    <xs:element name = "person" type = "PersonTyp" minOccurs = "0"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name = "OrganisationTyp">
  <xs:sequence>
    <xs:element name = "name" type = "xs:string"/>
    <xs:element name = "niederlassung" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "zusatz" type = "xs:string" minOccurs = "0"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name = "PersonTyp">
  <xs:sequence>
    <xs:element name = "anrede" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "titel" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "vorname" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "nachname" type = "xs:string"/>
    <xs:element name = "zusatz" type = "xs:string" minOccurs = "0"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name = "AdresseTyp">
  <xs:sequence>
    <xs:element name = "strasse" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "hausnummer" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "postfach" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "postfachleitzahl" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "postfachort" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "postleitzahl" type = "xs:string"/>
    <xs:element name = "ort" type = "xs:string"/>
    <xs:element name = "kreis" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "bundesland" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "land" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "zusatz" type = "xs:string" minOccurs = "0"/>
  </xs:sequence>
</xs:complexType>

<!-- Typen Erhebung und Berichtszeitrum -->

<xs:complexType name = "ErhebungTyp">
  <xs:sequence>
    <xs:element name = "kennung" type = "TextKlasseTyp"/>
    <xs:element name = "text" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "berechtigung" type = "TextKlasseTyp" minOccurs = "0"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name = "BerichtszeitraumTyp">
  <xs:choice>
    <xs:element name = "string" type = "StringTyp"/>
    <xs:group ref = "BerichtszeitraumGruppe"/>
  </xs:choice>
  <xs:sequence>
    <xs:element name = "beginn" type = "BeginnEndeTyp"/>
    <xs:element name = "ende" type = "BeginnEndeTyp"/>
  </xs:sequence>
</xs:complexType>

<xs:simpleType name="JahrTyp">
  <xs:restriction base="xs:positiveInteger">
    <xs:minInclusive value="0001"/>
    <xs:maxInclusive value="9999"/>
  </xs:restriction>

```

```

</xs:simpleType>

<xs:simpleType name = "HalbjahrTyp">
  <xs:restriction base = "xs:positiveInteger">
    <xs:minInclusive value = "1"/>
    <xs:maxInclusive value = "2"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name = "SemesterTyp">
  <xs:restriction base = "xs:positiveInteger">
    <xs:maxInclusive value = "2"/>
    <xs:minInclusive value = "1"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name = "QuartalTyp">
  <xs:restriction base = "xs:positiveInteger">
    <xs:maxInclusive value = "4"/>
    <xs:minInclusive value = "1"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name = "MonatTyp">
  <xs:restriction base = "xs:positiveInteger">
    <xs:maxInclusive value = "12"/>
    <xs:minInclusive value = "1"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name = "WocheTyp">
  <xs:restriction base = "xs:positiveInteger">
    <xs:maxInclusive value = "53"/>
    <xs:minInclusive value = "1"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name = "TagTyp">
  <xs:restriction base = "xs:positiveInteger">
    <xs:maxInclusive value = "31"/>
    <xs:minInclusive value = "1"/>
  </xs:restriction>
</xs:simpleType>

<xs:complexType name = "StringTyp">
  <xs:simpleContent>
    <xs:extension base = "xs:string">
      <xs:attribute name = "format" use = "required" type = "xs:normalizedString"/>
      <xs:attribute name = "klasse" use = "required">
        <xs:simpleType>
          <xs:restriction base = "xs:token">
            <xs:enumeration value = "undefiniert"/>
            <xs:enumeration value = "datum"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name = "BeginnEndeTyp">
  <xs:sequence>
    <xs:group ref = "BerichtszeitraumGruppe"/>
  </xs:sequence>
</xs:complexType>

<!-- Type Material -->

<xs:complexType name = "MaterialTyp">
  <xs:sequence>
    <xs:element name = "name" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "kennung" type = "TextKlasseTyp" minOccurs = "0"/>
    <xs:element name = "version" type = "xs:string" minOccurs = "0"/>
  </xs:sequence>
</xs:complexType>

```

```

<!-- Meldungs-Id und Meldungsart -->

<xs:complexType name = "MeldungsIDTyp">
  <xs:simpleContent>
    <xs:extension base = "TextKlasseNoDefaultTyp"/>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name = "MeldungsartTyp">
  <xs:choice>
    <xs:element name = "original"/>
    <xs:element name = "berichtigung" type = "FunktionselementTyp"/>
    <xs:element name = "nachmeldung" type = "FunktionselementTyp"/>
    <xs:element name = "austausch" type = "FunktionselementTyp"/>
    <xs:element name = "loeschung" type = "FunktionselementTyp"/>
    <xs:element name = "nil"/>
  </xs:choice>
</xs:complexType>

<xs:complexType name = "FunktionselementTyp">
  <xs:sequence>
    <xs:element name = "meldungsID" type = "MeldungsIDTyp"/>
  </xs:sequence>
</xs:complexType>

<!-- Typen Fehler, Nachricht, Meldung -->

<xs:simpleType name="SchluesselWertTyp">
  <xs:restriction base="xs:string"/>
</xs:simpleType>

<xs:complexType name = "SchluesselTyp">
  <xs:simpleContent>
    <xs:extension base="SchluesselWertTyp">
      <xs:attribute name="klasse" type="xs:string"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:simpleType name="GewichtWertTyp">
  <xs:restriction base="xs:string"/>
</xs:simpleType>

<xs:complexType name = "GewichtTyp">
  <xs:simpleContent>
    <xs:extension base="GewichtWertTyp">
      <xs:attribute name="klasse" type="xs:string"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:simpleType name="FormatPositionTyp">
  <xs:restriction base = "xs:token">
    <xs:enumeration value = "xpath"/>
    <xs:enumeration value = "byte"/>
    <xs:enumeration value = "char"/>
    <xs:enumeration value = "satz"/>
    <xs:enumeration value = "name"/>
  </xs:restriction>
</xs:simpleType>

<xs:complexType name = "PositionTyp">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="format" type="FormatPositionTyp"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name = "MerkmalTyp">
  <xs:simpleContent>
    <xs:extension base="xs:normalizedString">
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>

```

```

<xs:complexType name = "FehlerTyp">
  <xs:sequence>
    <xs:element name = "schluessel" type = "SchluesselTyp" minOccurs="0"/>
    <xs:element name = "gewicht" type = "GewichtTyp" minOccurs="0"/>
    <xs:element name = "text" type = "TextKlasseNoDefaultTyp" minOccurs="0"/>
    <xs:element name = "position" type = "PositionTyp" minOccurs="0"/>
    <xs:element name = "merkmal" type = "MerkmalTyp" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="klasse" type="xs:string" use="optional"/>
</xs:complexType>

<xs:complexType name = "NachrichtenIDTyp">
  <xs:simpleContent>
    <xs:extension base = "TextKlasseNoDefaultTyp"/>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name = "AbsenderTyp">
  <xs:sequence>
    <xs:element name = "kennung" type = "TextKlasseTyp" minOccurs = "1"/>
    <xs:element name = "berechtigung" type = "TextKlasseTyp" minOccurs = "0"/>
    <xs:element name = "identifikation" type = "IdentifikationKontaktTyp" minOccurs = "0"/>
    <xs:element name = "externeIdentifikation" type = "TextKlasseTyp" minOccurs = "0"/>
    <xs:element name = "kontakt" type = "IdentifikationKontaktTyp" minOccurs = "0"/>
    <xs:element name = "memo" type = "xs:string" minOccurs = "0"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name = "EmpfaengerTyp">
  <xs:sequence>
    <xs:element name = "kennung" type = "TextKlasseTyp" minOccurs = "0"/>
    <xs:element name = "identifikation" type = "IdentifikationKontaktTyp" minOccurs = "0"/>
    <xs:element name = "externeIdentifikation" type = "TextKlasseTyp" minOccurs = "0"/>
    <xs:element name = "kontakt" type = "IdentifikationKontaktTyp" minOccurs = "0"/>
    <xs:element name = "memo" type = "xs:string" minOccurs = "0"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name = "MeldungTyp">
  <xs:sequence>
    <xs:element name = "meldungsID" type = "MeldungsIDTyp" minOccurs = "0"/>
    <xs:element name = "meldungsart" type = "MeldungsartTyp" minOccurs = "0"/>
    <xs:group ref = "KontextGruppe" minOccurs = "1"/>
    <xs:element ref = "fehler" minOccurs = "0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="pruefstatus" type="PruefstatusTyp" use="required"/>
</xs:complexType>

<xs:complexType name = "NachrichtTyp">
  <xs:sequence>
    <xs:element name = "nachrichtenID" type = "NachrichtenIDTyp" minOccurs = "0"/>
    <xs:element ref = "fehler" minOccurs = "0" maxOccurs="unbounded"/>
    <xs:element ref = "meldung" minOccurs = "0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="pruefstatus" type="PruefstatusTyp" use="required"/>
</xs:complexType>

<xs:simpleType name="PruefstatusTyp">
  <xs:restriction base = "xs:token">
    <xs:enumeration value = "ungeprueft"/>
    <xs:enumeration value = "fehlerfrei"/>
    <xs:enumeration value = "fehlerhaft"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="DokumentstatusTyp">
  <xs:restriction base = "xs:token">
    <xs:enumeration value = "akzeptiert"/>
    <xs:enumeration value = "abgewiesen"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="PruefstufeWertTyp">
  <xs:restriction base="xs:integer">
    <xs:enumeration value = "1"/>
    <xs:enumeration value = "2"/>
  </xs:restriction>
</xs:simpleType>

```

```

        <xs:enumeration value = "3"/>
        <xs:enumeration value = "4"/>
        <xs:enumeration value = "5"/>
    </xs:restriction>
</xs:simpleType>

<xs:complexType name = "PruefstufeTyp">
    <xs:simpleContent>
        <xs:extension base="PruefstufeWertTyp">
            <xs:attribute name="klasse" type="xs:string"/>
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>

<xs:complexType name="DokumenttypTyp">
    <xs:sequence>
        <xs:element name="name" type="xs:string" minOccurs = "1"/>
        <xs:element name = "version" type = "VersionTyp" minOccurs = "0"/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="DokumentTyp">
    <xs:sequence>
        <xs:element name = "dateneingang" type = "DateneingangTyp" minOccurs = "0"/>
        <xs:element ref = "fehler" minOccurs = "0" maxOccurs="unbounded"/>
        <xs:element ref = "nachricht" minOccurs = "0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="pruefstatus" type="PruefstatusTyp" use="required"/>
    <xs:attribute name="dokumentstatus" type="DokumentstatusTyp" use="required"/>
</xs:complexType>

<xs:complexType name="SyntaxTyp">
    <xs:attribute name="pruefstatus" type="PruefstatusTyp" use="required"/>
</xs:complexType>

<xs:complexType name="SemantikTyp">
    <xs:attribute name="pruefstatus" type="PruefstatusTyp" use="required"/>
</xs:complexType>

<xs:complexType name="AutorisierungTyp">
    <xs:attribute name="pruefstatus" type="PruefstatusTyp" use="required"/>
</xs:complexType>

<xs:complexType name="DatenTyp">
    <xs:attribute name="pruefstatus" type="PruefstatusTyp" use="required"/>
</xs:complexType>

<xs:complexType name="PruefungTyp">
    <xs:sequence>
        <xs:element name = "pruefstufe" type="PruefstufeTyp" minOccurs = "0"/>
        <xs:element name = "dokumenttyp" type = "DokumenttypTyp" minOccurs = "0"/>
        <xs:element name = "syntax" type = "SyntaxTyp" minOccurs = "0"/>
        <xs:element name = "semantik" type = "SemantikTyp" minOccurs = "0"/>
        <xs:element name = "autorisierung" type = "AutorisierungTyp" minOccurs = "0"/>
        <xs:element name = "daten" type = "DatenTyp" minOccurs = "0"/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="PruefprotokollTyp">
    <xs:sequence>
        <xs:element name = "pruefung" type = "PruefungTyp" minOccurs = "1"/>
        <xs:element ref = "dokument" minOccurs = "1" maxOccurs="1"/>
    </xs:sequence>
</xs:complexType>

<!-- Gruppen -->

<xs:group name = "BerichtszeitraumGruppe">
    <xs:sequence>
        <xs:element name = "jahr" type="JahrTyp" minOccurs = "1"/>
        <xs:choice minOccurs = "0">
            <xs:element name = "halbjahr" type = "HalbjahrTyp"/>
            <xs:element name = "semester" type = "SemesterTyp"/>
            <xs:element name = "quartal" type = "QuartalTyp"/>
            <xs:element name = "woche" type = "WocheTyp"/>
        </xs:choice>
    </xs:sequence>
</xs:group>

```

```

        <xs:sequence>
          <xs:element name = "monat" type = "MonatTyp" minOccurs = "1"/>
          <xs:element name = "tag" type = "TagTyp" minOccurs = "0"/>
        </xs:sequence>
      </xs:choice>
    </xs:sequence>
  </xs:group>

  <xs:group name = "KontextGruppe">
    <xs:sequence>
      <xs:element ref = "erhebung" minOccurs = "1"/>
      <xs:element ref = "berichtszeitraum" minOccurs = "1"/>
      <xs:element ref = "berichtspflichtiger" minOccurs = "0"/>
      <xs:element ref = "berichtsempfaenger" minOccurs = "0"/>
    </xs:sequence>
  </xs:group>

  <!-- Top Level Elementtypdefinitionen
  Jedes dieser Elemente ist als Pseudo-Root validierbar
  -->

  <xs:element name = "anwendung" type = "AnwendungTyp"/>
  <xs:element name = "optionen" type="OptionenTyp"/>
  <xs:element name = "protokoll" type = "ProtokollTyp"/>
  <xs:element name = "absender" type = "AbsenderTyp"/>
  <xs:element name = "empfaenger" type = "EmpfaengerTyp"/>
  <xs:element name = "erhebung" type = "ErhebungTyp"/>
  <xs:element name = "berichtszeitraum" type = "BerichtszeitraumTyp"/>
  <xs:element name = "berichtspflichtiger" type = "AbsenderTyp"/>
  <xs:element name = "berichtsempfaenger" type = "EmpfaengerTyp"/>
  <xs:element name = "pruefprotokoll" type="PruefprotokollTyp"/>
  <xs:element name = "dokument" type = "DokumentTyp"/>
  <xs:element name = "nachricht" type = "NachrichtTyp"/>
  <xs:element name = "meldung" type = "MeldungTyp"/>
  <xs:element name = "fehler" type="FehlerTyp"/>

  <!-- Root Element
  -->

  <xs:element name = "DatML-RES-D">
    <xs:complexType>
      <xs:sequence>
        <xs:element name = "memo" type = "xs:string" minOccurs = "0"/>
        <xs:element ref = "optionen" minOccurs = "0"/>
        <xs:element ref = "protokoll" minOccurs = "0"/>
        <xs:element ref = "absender"/>
        <xs:element ref = "empfaenger"/>
        <xs:element ref = "pruefprotokoll"/>
      </xs:sequence>
      <xs:attribute name = "version" use = "required">
        <xs:simpleType>
          <xs:restriction base = "xs:token">
            <xs:enumeration value = "1.0"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:complexType>
  </xs:element>
</xs:schema>

```