

IT-Standards für die amtliche Statistik

Data Markup Language (DatML)

Komponente DatML/RAW 2.0

Ein XML-basierter Dokumenttyp für die Rohdatenerhebung

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

Status: Spezifikation

Stand: 11.01.2005

© Statistisches Bundesamt Wiesbaden, Deutschland

Data Markup Language/RAW (DatML/RAW) 2.0

Spezifikation

Beschreibung

DatML ist ein XML-basierter Dokumenttyp für statistische Daten; er 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, und für die sichere, anwendungs- und plattformneutrale Archivierung.
- DatML/ASK für die Beschreibung von Fragebögen.

Im vorliegenden Dokument ist der Dokumenttyp DatML/RAW beschrieben.

Dokumentstatus

Dieses Dokument ist die verbindliche Spezifikation des Dokumenttyps DatML/RAW in der Version 2.0.

Version	Datum	Autor	Stand/Änderung
1	02.12.2004	Schäfer	Freigabe
2	10.01.2005	Schäfer	Austausch Schema

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.

Kompatibilität

Version 2.0 ist inkompatibel mit Version 1.0.2 und deren Vorgängerversionen. Die Inkompatibilitäten sind in Abschnitt 4.1.1 beschrieben.

Änderungen in der Version 2.0

Die den Dokumenttyp betreffenden Änderungen sind in Kapitel 4 dokumentiert. Weitere Änderungen dieser Spezifikation sind redaktioneller Natur.

Die wichtigsten Änderungen seit Version 1.0.2 sind

- Folgende Komponenten eines DatML/RAW-Dokument können nun mit einer ID versehen werden:
 - Das gesamte Dokument (Elementtyp <dokumentID>)
 - Jede Nachricht (Elementtyp <nachrichtenID>)
 - Jede Meldung (Elementtyp <meldungsID>)
- Externe Erzeuger von DatML/RAW-Dokumenten, die eigene Identifizierungsverfahren für am Lieferverfahren Beteiligte wie ihre Kunden und die statistischen Ämter verwenden, können einem geeigneten Schlüssel ö.ä. mit Hilfe des Elementtypes <externeIdentifikation> bei jedem Beteiligten ablegen.
- Für die Protokollierung von Dateneingang und Dokumentinstanzen wurden neue Strukturen entwickelt, die Elemente anstelle von Attributen verwenden; die Vorteile sind bessere Validierbarkeit und Erweiterbarkeit.
- Merkmalsgruppen können geschachtelt werden.
- Semesterangaben im Berichtszeitraum.

1. Inhaltsverzeichnis

1. Inhaltsverzeichnis	4
2. Einführung	8
2.1. Versionierung, Namensraum, Schemata	8
2.2. Verwendung	9
2.2.1. Ein generischer Dokumenttyp	9
2.2.2. Anwendungen von DatML/RAW	9
2.2.2.1. Spezifikation einer Anwendung	9
2.2.2.2. Softwareanforderungen, Schichtenmodell [<i>nicht-normativ</i>]	10
2.2.3. Verfügbarkeit	10
2.3. Struktur	11
2.3.1. Nachrichten	11
2.3.2. Segment, Segmentbaum	11
2.3.3. Datensegment, Segmentpfad	11
2.3.4. Statistischer Kontext	12
2.3.5. Meldungen	12
2.4. Metadaten	13
2.4.1. Absender und Empfänger	13
2.4.2. Berichtspflichtiger und Berichtsempfänger	13
2.4.3. Identifikation	13
2.4.4. Externe Identifikation	14
2.4.5. Berechtigung	14
2.4.6. Adressangaben	14
2.4.7. Kontaktinformationen	14
2.4.8. Korrekturen	15
2.4.9. Verarbeitungsoptionen	15
2.4.9.1. Testunterstützung	15
2.4.9.2. Empfangsbestätigung	15
2.4.10. Protokoll	15
2.4.11. Identifikation des Dokumentes	16
2.5. Daten (Statistische Informationen)	16
2.5.1. Datensätze	16
2.5.2. Merkmale (Statistische Variablen)	17
2.5.2.1. Einfache Merkmale	17
2.5.2.2. Ordnungsmerkmale	17
2.5.2.3. Hilfsmerkmale	18
2.5.3. Merkmalsgruppen	18
2.5.3.1. Identifikation und Indizierung	18
2.5.3.2. Geschachtelte Merkmalsgruppen	21
2.5.4. Namen	21
2.5.4.1. Wertebereich	21
2.5.4.2. Namenskonflikte	21
2.5.5. Sortierung	22
2.5.6. Numerische Werte	22
2.5.7. Formatierte Werte	24
2.5.7.1. Klasse <code>datum</code>	24
2.6. Verarbeitung [<i>nicht-normativ</i>]	26
2.6.1. Verteilung	26
2.6.2. Zerlegung	26
2.7. Weiteres	28

2.7.1. Informationen aus Online-Erhebungen	28
2.7.2. Unterstützung zeilenbasierter Formate	28
2.7.2.1. Zeilenbasierte Ablage	28
2.7.2.2. Abbildung von Typen auf Zeilen	29
2.7.3. Vorgeschriebene Elemente	30
2.7.4. Einfügen von Daten	30
3. Elementtypen.....	31
3.1. Darstellung der Elementtypen und Attribute	32
3.1.1. Graphische Darstellung der Elementtypen	32
3.1.2. Tabellarische Darstellung der Elementtypen	32
3.1.3. Tabellarische Darstellung der Attribute	33
3.2. Dokument, Nachricht, Segment.....	34
3.2.1. Element <DatML-RAW-D>.....	34
3.2.2. Element <nachricht>	35
3.2.3. Element <nachrichtenID>	36
3.2.4. Element <segment>	37
3.3. Verfahrensteilnehmer, Identifikation, Kontaktangaben	39
3.3.1. Element <absender>	39
3.3.2. Element <adresse>	40
3.3.3. Element <anrede>	41
3.3.4. Element <berichtsempfaenger>.....	42
3.3.5. Element <berichtspflichtiger>	43
3.3.6. Element <bundesland>.....	44
3.3.7. Element <email>	45
3.3.8. Element <empfaenger>.....	46
3.3.9. Element <externeIdentifikation>	47
3.3.10. Element <fax>.....	48
3.3.11. Element <hausnummer>.....	49
3.3.12. Element <identifikation>	50
3.3.13. Element <identitaet>.....	51
3.3.14. Element <kontakt>	52
3.3.15. Element <korrektur>	53
3.3.16. Element <kreis>	54
3.3.17. Element <land>.....	55
3.3.18. Element <nachname>	56
3.3.19. Element <niederlassung>	57
3.3.20. Element <organisation>	58
3.3.21. Element <ort>.....	59
3.3.22. Element <person>	60
3.3.23. Element <postfach>	61
3.3.24. Element <postfachleitzahl>	62
3.3.25. Element <postfachort>.....	63
3.3.26. Element <postleitzahl>	64
3.3.27. Element <strasse>	65
3.3.28. Element <telefon>	66
3.3.29. Element <titel>	67
3.3.30. Element <url>.....	68
3.3.31. Element <vorname>	69
3.3.32. Element <zeile>	70

3.4. Erhebung, Berichtszeitraum	71
3.4.1. Elemente <beginn>, <ende>	71
3.4.2. Element <berichtszeitraum>	72
3.4.3. Element <erhebung>	74
3.4.4. Elemente <jahr>, <halbjahr>, <semester>, <quartal>, <monat>, <woche>, <tag>	75
3.5. Organisatorische und technische Informationen	76
3.5.1. Element <datenattribute>	76
3.5.2. Element <material>	77
3.6. Datensegment	78
3.6.1. Elemente <austausch>, <berichtigung>, <loeschung>, <nachmeldung>	78
3.6.2. Element <datensegment>	79
3.6.3. Element <hmm>	80
3.6.4. Element <meldungsart>	81
3.6.5. Element <meldungsID>	82
3.6.6. Element <mm>	83
3.6.7. Element <mmgr>	84
3.6.8. Element <na>	85
3.6.9. Elemente <nil>, <original>	86
3.6.10. Element <omm>	87
3.6.11. Element <satz>	89
3.6.12. Element <wert>	90
3.7. Verarbeitungssteuerung	91
3.7.1. Element <empfangsbestaetigung>	91
3.7.2. Element <optionen>	92
3.7.3. Element <test>	93
3.8. Protokollierung	94
3.8.1. Übersicht	94
3.8.2. Element <anwendung>	95
3.8.3. Element <anwendungsname>	96
3.8.4. Element <datei>	97
3.8.5. Element <dateiname>	98
3.8.6. Element <dateneingang>	99
3.8.7. Element <datenformat>	101
3.8.8. Element <datentraeger>	102
3.8.9. Element <dokumentID>	103
3.8.10. Element <dokumentinstanz>	104
3.8.11. Element <eingangsstempel>	105
3.8.12. Element <formularID>	106
3.8.13. Element <formularname>	107
3.8.14. Element <hersteller>	108
3.8.15. Element <onlineformular>	109
3.8.16. Element <papierformular>	110
3.8.17. Element <protokoll>	111
3.8.18. Element <ressourceID>	112
3.8.19. Element <server>	113
3.8.20. Element <upload>	114

3.8.21. Element <zertifikat>.....	115
3.9. Sonstige Elemente.....	116
3.9.1. Element <anzahl>	116
3.9.2. Element <berechtigung>	117
3.9.3. Element <datum>	118
3.9.4. Element <kennung>	119
3.9.5. Element <memo>.....	120
3.9.6. Element <name>.....	121
3.9.7. Element <string>	122
3.9.8. Element <text>.....	123
3.9.9. Element <uhrzeit>	124
3.9.10. Element <version>	125
3.9.11. Element <zusatz>	126
4. Änderungen gegenüber Version 1.0.2.....	127
4.1. Kompatible Änderungen	127
4.1.1. Dokument, Nachricht, Segment.....	127
4.1.2. Teilnehmer, Identifikation, Kontaktangaben.....	127
4.1.3. Erhebung, Berichtszeitraum	127
4.1.4. Datensegment	128
4.1.5. Sonstige Elemente.....	128
4.2. Inkompatible Änderungen.....	128
4.2.1. Dokument, Nachricht, Segment.....	128
4.2.2. Protokollierung.....	128
4.2.3. Daten	129
4.3. Sonstige Änderungen	129
4.3.1. Allgemein.....	129
4.3.2. Protokollierung.....	129
5. Beispieldokument [<i>nicht-normativ</i>].....	130
6. DTD und XML-Schema 2.0 [<i>nicht-normativ</i>].....	141
6.1. DTD	141
6.2. XML-Schema.....	151

2. Einführung

2.1. Versionierung, Namensraum, Schemata

Die Versionsbezeichnung für DatML/RAW 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 englische Version von DatML/RAW 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.

Achtung: Das Versionierungsschema wurde erst mit Version 2.0 eingeführt. Für die Versionen 1.x ergeben sich Abweichungen.

Version 2.x:

Namensraum	http://www.destatis.de/schema/datml-raw/version.0/lang
Schema	<code>datml-raw-lang-version_0.dtd</code> <code>datml-raw-lang-version_0.xsd</code>

Version 1.x:

Namensraum	http://www.destatis.de/schema/datml-raw/1.0/d http://www.destatis.de/schema/datml-raw/1.0/e
Schema	<code>datml-raw-1_0.dtd</code> <code>datml-raw-1_0.xsd</code>

Anhand dieser Namenschemata ergeben sich folgende Namensräume und Schemadateinamen

Version	Sprache / Schematyp	Namensraum / Schema
---------	---------------------	---------------------

1.x	Deutsch	http://www.destatis.de/schema/datml-raw/1.0/d
	Englisch	http://www.destatis.de/schema/datml-raw/1.0/e
	DTD	datml-raw-1.dtd
	XSD	datml-raw-1.xsd
2.x	Deutsch	http://www.destatis.de/schema/datml-raw/2.0/de
	Englisch	http://www.destatis.de/schema/datml-raw/2.0/en
	DTD	datml-raw-de-2_0.dtd, datml-raw-en-2_0.dtd
	XSD	datml-raw-de-2_0.xsd, datml-raw-en-2_0.xsd

2.2. Verwendung

2.2.1. Ein generischer Dokumenttyp

DatML/RAW ist ein XML-Dokumenttyp für die Lieferung statistischer Rohdaten und Metadaten. Aus fachlicher – statistischer – Sicht ist DatML/RAW ein *generischer* Dokumenttyp, da er erhebungsübergreifend verwendet werden kann. Seine Struktur ist auf größtmögliche Flexibilität ausgelegt und erlaubt die Lieferung von Daten für beliebige und beliebig viele Erhebungen, für unterschiedliche Berichtszeiträume, von unterschiedlichen Berichtspflichtigen und für unterschiedliche Berichtsempfänger in *einem* Dokument. DatML/RAW ist sowohl absender- wie auch empfängerseitig *mandantenfähig*. Desweiteren ermöglicht DatML/RAW *Zusatzfunktionen* wie Testunterstützung, Adressänderung und Empfangsbestätigung.

2.2.2. Anwendungen von DatML/RAW

Die Lieferung von statistischen Rohdaten im DatML/RAW-Format für eine bestimmte Erhebung mit ihrem definierten Umfang statistischer Merkmale ist eine *Anwendung des generischen Dokumenttyps* DatML/RAW. Abgesehen von den statistischen Merkmalen kann jede Anwendung spezifische Abweichungen vom generischen Dokumenttyp in Form funktionaler und struktureller Einschränkungen aufweisen. Die wesentlichen Gründe sind vor allem organisatorische (z.B. keine Meldung durch Dritte) und technische (z.B. fehlende Softwareunterstützung).

HINWEIS: Es ist möglich, für eine Anwendung von DatML/RAW einen abgeleiteten Dokumenttyp in Form einer DTD oder eines entsprechenden anderen Schemas zu definieren, der zusätzliche anwendungsspezifische Restriktionen enthält. Unabhängig von technischen Aspekten muss jede DatML/RAW-Anwendung sicherstellen, dass ein abgeleiteter Dokumenttyp vollkommen konform mit dem vollständigen Dokumenttyp ist und alle fehlerfreien Dokumente dieser Anwendung gegen beide Varianten fehlerfrei validierbar sind.

2.2.2.1. Spezifikation einer Anwendung

Alle Eigenschaften einer DatML/RAW-Anwendung müssen spezifiziert werden und ergeben ein *Anwendungsprofil*, das folgende Bereiche abdecken muss:

- Beschreibung funktionaler und struktureller Einschränkungen, insbesondere

- Zulässigkeit der Lieferung durch Dritte bzw. des Empfangs für Dritte (sogeannte Mandantenfähigkeit);
- Unterstützung von Verarbeitungsoptionen (s. 2.4.9);
- Unterstützung von Adresskorrekturen etc.;
- Zulässigkeit der Lieferung für mehrere Erhebungen bzw. Berichtszeiträume;
- Zulässigkeit der Verwendung von Ordnungsmerkmalen.
- Identifikation der Absender bzw. der Berichtspflichtigen
- Festlegung der statistischen Merkmale und ggf. Merkmalsgruppen sowie der erlaubten Merkmalsausprägungen (Werte).

2.2.2.2. Softwareanforderungen, Schichtenmodell [*nicht-normativ*]

Aus einem DatML/RAW-Anwendungsprofil ergibt sich ein individuelles Anforderungsprofil für die Verarbeitungssoftware. Das Erstellen maßgeschneiderter, mit einem Anforderungsprofil abgestimmter Lösungen bedeutet aber den Verlust der Vorteile generischer Programmierung. Eine Möglichkeit, dies zu mildern, ist die Verteilung der Dokumentverarbeitung auf mehrere generische Schichten: eine Empfangs- und Verteilungsschicht, eine Anwendungsschicht und eine Rückmeldungsschicht.

- Die **Empfangs- und Verteilungsschicht** würde ein Dokument empfangen, Absender, Empfänger, Berichtspflichtige und Berichtsempfänger identifizieren, ggf. prüfen und das Lieferdokument in DatML/RAW-Einzellieferungen zerlegen. Im Fall des Empfangs durch Dritte würde ggf. eine zweistufige Verteilung stattfinden: zunächst an die Berichtsempfänger und dort in Einzellieferungen (nach Erhebung, Berichtszeitraum und Berichtspflichtigen) an die Anwendungsschicht. Die Empfangs- und Verteilungsschicht wäre in diesem Fall funktional und physisch auf mehrere Komponenten und ggf. Installationsorte verteilt. Alle Funktionen der Empfangs- und Verteilungsschicht stünden aber in beiden Fällen anwendungsübergreifend zur Verfügung (s. 2.6.1).
- Aufgabe der **Anwendungsschicht** wäre die erhebungsspezifische Verarbeitung der Einzellieferungen, z.B. die Abbildung auf andere Dateiformate und die Plausibilisierung im Fachverfahren, basierend auf externen Beschreibungsobjekten (z.B. DatML/SSP, DatML/SDF).
- Die **Rückmeldungsschicht** wäre für die Rückmeldung an den Absender des Lieferdokumentes zuständig; sie würde aus der Empfangs- und der Anwendungsschicht Dokumente im Format DatML/RES empfangen und verarbeiten, vollständig automatisiert bis hin zum Versand maschineller Quittungen oder in Form der Vorbereitung und Veranlassung manueller Vorgänge (Ausdruck, Postversand).

2.2.3. Verfügbarkeit

DatML/RAW 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.

2.3. Struktur

Jedes DatML/RAW-Dokument muss folgende Komponenten enthalten: den Absender (<absender>), den Empfänger (<empfaenger>) und *mindestens eine* Nachricht (<nachricht>, s.2.3.1). Jede Nachricht kann *Segmente* unterteilt sein (<segment>, s.2.3.2).

Die Unterteilung eines Dokumentes in Nachrichten und einer Nachricht in Segmente ist ein flexibler Mechanismus für die Verteilung und Zuordnung von Daten und Metadaten. So kann ein Absender, der Daten für mehrere Berichtspflichtige übermittelt, die Daten eines jeden zusammenhängend als Nachricht ablegen; ein anderer Absender, der nur eigene Daten übermittelt, kann Nachrichten für die Trennung von Berichtszeiträumen verwenden (z.B. eine Nachlieferung, die mit den aktuellen Daten geliefert wird). Aber auch Nachrichten können Daten beliebiger statistischer Kontexte in separaten Segmenten enthalten.

2.3.1. Nachrichten

Nachrichten enthalten statistische Rohdaten sowie Metadaten, die den *statistischen Kontext* einer Rohdatenlieferung beschreiben. Die optionale Segmentierung einer Nachricht erlaubt es, beliebig viele Rohdatenlieferungen mit unterschiedlichem statistischen Kontext in einer Nachricht zu übermitteln und die Metadaten redundanzfrei abzulegen.

2.3.2. Segment, Segmentbaum

Ein Segment kann entweder weitere Segmente oder Datensätze enthalten. Durch die hierarchische Anordnung von Segmentknoten entsteht ein sogenannter *Segmentbaum*. Die Wurzel des Segmentbaumes ist der Nachrichtenknoten (<nachricht>), der die gleichen Eigenschaften wie ein Segmentknoten hat. Der Nachrichtenknoten kann daher logisch wie ein Segmentknoten behandelt werden. Die Blätter des Segmentbaumes sind die Segmentknoten, die anstelle weiterer Segmentknoten ein *Datensegment* (2.3.3) enthalten. In unsegmentierten Nachrichten besteht der Segmentbaum nur aus dem Nachrichtenknoten, der zugleich Wurzel- und Blattknoten ist.

2.3.3. Datensegment, Segmentpfad

Ein Blatt eines Segmentbaumes enthält wiederum mindestens ein Metadatum und – anstelle weiterer Segmentknoten – ein sogenanntes *Datensegment* (<datensegment>). Ein Datensegment ist der Abschnitt eines Segmentes, der die Meldungsdatensätze (<satz>) enthält, welche Ordnungsmerkmalen (<omm>) nachgeordnet sein können.

Ein *Segmentpfad* ist der Weg von der Wurzel des Segmentbaumes bis zu einem Blatt bzw. einem Datensegment (In unsegmentierten Nachrichten ist der Nachrichtenknoten Wurzel und Blatt zugleich). Die Summe der in einem Segmentpfad abgelegten Metadaten muss mindestens die nicht-optionalen Metadaten des *statistischen Kontextes* enthalten (s. 2.3.4). Eine semantische Überlappung ist nicht erlaubt, d.h. jedes Metadatum darf innerhalb einer Nachricht auf einem einzelnen Segmentpfad

nur ein einziges Mal vorkommen. Enthält ein Segmentknoten mehr als ein Metadatum, muss die vorgeschriebene Reihenfolge beachtet werden.

Die Metadaten des statistischen Kontextes erlauben es, in jedem Datensegment einen in sich abgeschlossenen Rohdatenbestand zu liefern. Grundsätzlich entspricht daher ein Datensegment einer Rohdatenlieferung.

HINWEIS: Es ist nicht ausgeschlossen, dass in einem DatML/RAW-Dokument zwei oder mehr Datensegmente, ggf. in verschiedenen Nachrichten, denselben statistischen Kontext haben, oder mit anderen Worten, dass mehrere Segmentpfade einen identischen statistischen Kontext liefern. Diese Spezifikation verzichtet auf eine Regelung für diesen Fall und überlässt es dem jeweiligen Anwenderkreis, dies für zulässig zu erklären oder nicht.

2.3.4. Statistischer Kontext

Innerhalb eines Datensegmentes haben alle Daten denselben statistischen Kontext. Die den statistischen Kontext beschreibenden Metadaten sind Kindknoten des Elementtyps `<nachricht>` und/oder des Elementtyps `<segment>`, sie sind also entlang der Segmentpfade angeordnet, die zu den Datensegmenten führen (s. 2.3.3). Die Reihenfolge der Anordnung der Metadaten auf dem Segmentpfad ist beliebig. Es darf jedoch keine semantische Überlappung geben, d.h. ein Metadatum darf in jedem Segmentpfad nur ein einziges Mal vorhanden sein, die notwendige Prüfung muss durch die Anwendung erfolgen.

Metadaten des statistischen Kontextes sind:

Metadatum	Elementtyp	Angabe	Default
Berichtsempfänger	<code><berichtsempfaenger></code>	optional	Empfänger
Berichtspflichtiger	<code><berichtspflichtiger></code>	optional	Absender
Berichtszeitraum	<code><berichtszeitraum></code>	muss	-
Datenattribute	<code><datenattribute></code>	optional	-
Erhebung	<code><erhebung></code>	muss	-
Hilfsmerkmal	<code><hmm></code>	optional	-
Materialbezeichnung	<code><material></code>	optional	-

2.3.5. Meldungen

Um die Übertragung von Daten Auskunftgebender an die Statistik zu beschreiben, werden häufig die Begriffe *Meldung* und *Einzelmeldung* gebraucht. Diese Begriffe sind nicht formal definiert und werden daher oft unterschiedlich interpretiert. Um Missverständnissen bei der Übertragung dieser Begriffe in den Kontext dieser Spezifikation vorzubeugen, wird für diese der Begriff *Meldung* wie folgt definiert:

Definition: Eine *Meldung* im Sinne dieser Spezifikation besteht aus einem Datensegment (s. 2.3.3) und allen Metadaten, die sich auf dem Segmentpfad vom über-

geordneten Nachrichtenknoten (s. 2.3.1) zu diesem Datensegment befinden, einschließlich der Metadaten, die ggf. durch Metadaten aus anderen Strukturen substituiert werden (das sind der Berichtspflichtige und der Berichtsempfänger, die bei Fehlen durch den Absender bzw. den Empfänger substituiert werden).

2.4. Metadaten

DatML/RAW erfordert und ermöglicht es, außer den statistischen Erhebungsdaten zusätzliche Informationen zu übermitteln:

- Technische Beschreibungsdaten wie Lieferweg, Erstellungsdatum;
- Funktionsorientierte Daten wie Verarbeitungsoptionen;
- Organisatorische Informationen wie Absender, Empfänger;
- Statistische, merkmalsübergreifende Daten wie Erhebung, Berichtszeitraum;
- Beschreibungsdaten zu einzelnen statistischen Merkmalen, wie Name, Einheit.

Diese beschreibenden bzw. funktionalen Informationen werden als Metadaten bezeichnet und dadurch von den Daten bzw. statistischen Werten unterschieden (s. 2.4.11). Ohne maschinell verarbeitbare Metadaten sind weitere Steigerungen der Datenqualität und der Genauigkeit der statistischen Ergebnisse nur mit extrem hohen Aufwendungen erreichbar. Darüber hinaus finden Metadaten Verwendung in Neben- und Unterstützungsprozessen wie Identifikation, Adressverwaltung und Plausibilisierung und sind für die Automatisierung statistischer Erhebungs- und Aufbereitungsprozesse und die Verbesserung statistischer Dienstleistungen unerlässlich.

2.4.1. Absender und Empfänger

Jedes DatML/RAW-Dokument muss genau einen Absender und einen Empfänger angeben. Der Absender versendet das Dokument; er kann selbst Berichtspflichtiger sein und/oder für Dritte berichten. Für Änderungen der Adress- und Kontaktinformationen des Absenders steht der Knoten `<korrektur>` zur Verfügung. Der Empfänger ist der Adressat des Dokumentes, er ist aber nicht notwendigerweise auch Adressat der Daten, also Berichtsempfänger, sondern kann auch im Namen Dritter Dokumente empfangen.

2.4.2. Berichtspflichtiger und Berichtsempfänger

Es können Daten beliebig vieler Berichtspflichtiger an beliebig viele Berichtsempfänger (die erhebenden Stellen) in einem Dokument übermittelt werden. Standardmäßig sind der Berichtspflichtige mit dem Absender und der Berichtsempfänger mit dem Empfänger des Dokumentes identisch. Berichtspflichtige und -empfänger müssen daher nur angegeben werden, wenn sie nicht Absender bzw. Empfänger des Dokumentes sind. (s. 2.3.4). Für Änderungen der Adress- und Kontaktinformationen eines Berichtspflichtigen steht der Elementtyp `<korrektur>` zur Verfügung.

2.4.3. Identifikation

Die Identifikation der an einem Lieferverfahren Beteiligten – s. 2.4.1 und 2.4.2 – kann allein mit Hilfe einer Kennung (`<kennung>`), deren Angabe für den Absender und

den Berichtspflichtigen vorgeschrieben ist und mit einem Berechtigungsschlüssel kombiniert werden kann (s. 2.4.3), aber auch unter Verwendung von Namens- und Adressangaben erfolgen (<identifikation>). Hierfür stehen wahlweise ein typbasiertes (<identitaet>, <adresse>) und ein zeilenbasiertes Modell (<zeile>) zur Verfügung. Das verwendete Identifikationsverfahren muss den Umfang der notwendigen Informationen und Prüfungen festlegen und ist der Regelung der Anwender überlassen.

2.4.4. Externe Identifikation

Externe Erzeuger von DatML/RAW-Dokumenten benutzen in vielen Fällen eigene Identifizierungsverfahren für die an einem Lieferverfahren Beteiligten – s. 2.4.1 und 2.4.2 –, wie z.B. Kundennummern. DatML/RAW erlaubt bei jedem Beteiligten einen zusätzlichen Schlüssel o.ä. abzulegen, der für externe Identifizierungszwecke verwendet werden kann. Hierzu dient der Elementtyp <externeIdentifikation>, der in die Inhaltsmodelle der entsprechenden Elementtypen als neues optionales Kindelement aufgenommen wurde.

2.4.5. Berechtigung

Für den Nachweis der Berechtigung des Absender bzw. Berichtspflichtigen zur Teilnahme an einem Lieferverfahren kann ein Berechtigungsschlüssel verlangt werden. Hierfür steht in der Beschreibung des Absenders bzw. Berichtspflichtigen der Elementtyp <berechtigung> zur Verfügung. Da ein Berechtigungsschlüssel erhebungsabhängig sein kann, ist er auch im Inhaltsmodell des Elementtyps <erhebung> zulässig.

Regel: Ist ein Berechtigungsschlüssel für einen Berichtspflichtigen über alle Erhebungen einheitlich, genügt es, ihn beim Berichtspflichtigen (bzw. dem Absender) anzugeben. Ist der Berechtigungsschlüssel *außerdem* erhebungsspezifisch, muss er, *wenn ein Berichtspflichtiger für mehr als ein Erhebung meldet*, bei der Erhebung angegeben werden, es sein denn, die Metadaten sind so organisiert, dass es keine strukturelle Unterordnung der Erhebung unter den Berichtspflichtigen gibt; dies ist allerdings ohne Redundanz nicht möglich.

2.4.6. Adressangaben

Einige der in typbasierten Adressangaben (<adresse>) erlaubten Bestandteile werden in Deutschland üblicherweise nicht benutzt, z.B. <kreis>; diese Angaben dienen vor allem der Symmetrie zur englischen Version von DatML/RAW. Grundsätzlich sollten Adressen immer so vollständig wie möglich angegeben werden. Für Korrekturen wird ein identisches Inhaltsmodell verwendet. In beiden Fällen können Mussangaben (PLZ, Ort) als leere Elemente abgelegt werden, wenn keine Daten für sie vorhanden sind, aber ein gültiges Dokument erzeugt werden soll.

2.4.7. Kontaktinformationen

Kontaktinformationen beschreiben den Zugang zu Ansprechpartnern bei den Verfahrensbeteiligten. Analog zur Identifikation stehen wahlweise ein typbasiertes (<iden-

titaet>, <telefon>, <email>, <fax>) und ein zeilenbasiertes Modell (<zeile>) zur Verfügung.

2.4.8. Korrekturen

Identifikations- und Kontaktinformationen eines Absenders oder Berichtspflichtigen können über den Elementtyp <korrektur> aktualisiert werden. Die Details eines Korrekturverfahrens sind anwendungsabhängig und in dieser Spezifikation nicht enthalten.

2.4.9. Verarbeitungsoptionen

Verarbeitungsoptionen geben dem Absender eines DatML/RAW-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>.

2.4.9.1. Testunterstützung

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

2.4.9.2. Empfangsbestätigung

Der Elementtyp <empfangsbestaetigung> erlaubt die Anforderung einer Empfangsbestätigung. Optionale Attribute bestimmen Versandart, Format und eine Betreffangabe. Die Empfangsbestätigung kann über die Bestimmung der Versandart explizit als elektronisches (*versandart*="email") oder als Papierdokument (*versandart*="post") angefordert werden; für elektronische Dokumente ist die Auswahl des Formates möglich, nämlich als angehängte XML-Datei im Format DatML/RES (*format*="xml") oder als generierte Textdatei (*format*="text"). Der Standardwert für Versandart und Format ist jeweils "default", d.h., bei Abwesenheit expliziter Werte werden diese von der Anwendung gesetzt.

2.4.10. Protokoll

Protokollinformationen (<protokoll>) dienen dazu, Lieferwege und Verarbeitungsstufen eines DatML/RAW-Dokumentes zu dokumentieren. Der Eingang der Rohdaten (<dateneingang>) und die Dokumentinstanzen (<dokumentinstanz>) werden separat protokolliert. Für den Dateneingang können Details wie Lieferweg, Lieferdatenträger und Lieferformat festgehalten werden, was es erlaubt, zwischen Online-Meldungen per HTML-Formular, Papiermeldungen, dem Upload von Dateien und Datenträgerlieferungen zu unterscheiden.

Protokollinformationen sind optional; für die Befüllung der Protokollstrukturen gelten folgende Regeln:

Regel: Die Anwendung bzw. Komponente, die den Dateneingang bearbeitet, dokumentiert Datum und Uhrzeit, den Eingangstempel sowie sich selbst im Element `<dateneingang>`.

Regel:

- Dateilieferungen werden im Element `<datei>` dokumentiert.
- Dateiname, Datum, Uhrzeit und Anwendung beziehen sich auf die Erstellung der Lieferdatei beim *Absender*.
- Die Dokument-ID ist zu belegen, wenn eine entsprechende Angabe aus der Lieferdatei entnommen werden kann.
- Das Datenformat und Datenträgerinformationen bzw. ein Datei-Upload sind zu dokumentieren, wenn die Anwendung diese Informationen ermitteln kann.

Regel:

- Dokumentinstanzen sind in der Reihenfolge ihrer Entstehung zu protokollieren, d.h. in einer Folge von Elementen des Typs `<dokumentinstanz>` beschreibt das erste die erste und das letzte die aktuelle Instanz des Dokumentes.
- Wird ein DatML/RAW-Dokument geliefert, welches ein Protokoll enthält, 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.

2.4.11. Identifikation des Dokumentes

Es ist möglich, für jede Instanz eines DatML/RAW-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, insbesondere für die Identifikation von Dokumenten (Nachrichten), die zwischen den Meldern und den erhebenden Stellen ausgetauscht werden. Das Format der Identifizierungskennung ist anwendungsabhängig und darf die Wohlgeformtheit des Dokumentes nicht beeinträchtigen.

2.5. Daten (Statistische Informationen)

Der Begriff *Daten* im Sinne dieser Spezifikation umfasst die statistischen Merkmale, für die Werte von den Berichtspflichtigen erhoben werden. Daten sind quasi die reine statistische Nutzlast, statistisch verwertbare Informationen über Erhebungsobjekte.

2.5.1. Datensätze

Ein statistischer Datensatz stellt eine Informationseinheit dar, die zusammenhängende Daten über ein Erhebungsobjekt enthält. In diesem Sinne können Datensätze auch innerhalb DatML/RAW verwendet werden. Die dokumenttypbezogene Funktion eines DatML/RAW-Datensatzes (`<satz>`) ist die eines Gültigkeitsbereiches für statistische Merkmale; sie müssen innerhalb eines Datensatzes eindeutig sein (s. 2.5.2).

2.5.2. Merkmale (Statistische Variablen)

Daten werden Merkmalen als Wert zugeordnet (`<wert>`) und durch Attribute beschrieben. Es gibt zwei Arten von Merkmalen: *einfache Merkmale*, die Bestandteil von Datensätzen sind, und die den Datensätzen übergeordneten *Ordnungsmerkmale*. In beiden Fällen ist das wichtigste Attribut eines Merkmals der *Merkmalsname*, der es identifiziert.

2.5.2.1. Einfache Merkmale

Ein einfaches Merkmal (`<mm>`) hat einen Namen (Attribut `name`) und einen Wert (`<wert>`). Ein Wert kann durch Attribute näher beschrieben werden, er kann leer oder durch eine explizite Nichtantwort (`<na>`) ersetzt sein. *Merkmalsgruppen* (s. 2.5.3) fassen Merkmale zu einer Gruppe zusammen und sind indizier- und wiederholbar. Datensätze und Merkmalsgruppen wirken als Sichtbarkeitsbereiche für einfache Merkmale, innerhalb denen jedes Merkmal bzw. dessen Name eindeutig sein muss. Mehrere Instanzen eines Merkmals in einem Datensatzes sind nur dann möglich, wenn das Merkmal Bestandteil einer Merkmalsgruppe ist.

Regel: In einem Meldungsdatensatz darf ein Merkmal nur dann mehr als einmal vorkommen, wenn es Bestandteil einer Merkmalsgruppe ist.

2.5.2.2. Ordnungsmerkmale

Ordnungsmerkmale (`<omm>`) haben wie einfache Merkmale einen Namen (Attribut `name`) und einen Wert (`<wert>`), der jedoch nicht fehlen und nicht leer sein darf. Im Ggs. zu einfachen Merkmalen handelt es sich dabei um Merkmale, die eine systematische – oder eben ordnende – Aussage haben. In Verzeichnissen geführte Merkmale wie Länderschlüssel und Betriebsnummern sind typische Ordnungsmerkmale.

Mit Ordnungsmerkmalen können Gruppen von Datensätzen gebildet werden, die alle denselben Wert für ein bestimmtes Ordnungsmerkmal aufweisen. Dieses Ordnungsmerkmal ist in der Dokumentstruktur nicht Bestandteil der Datensätze, sondern den Datensätzen übergeordnet. Eine Folge von Datensätzen wird jeweils genau einer Ausprägung – einem Wert – eines Ordnungsmerkmals (und ggf. denen übergeordneter Ordnungsmerkmale) zugeordnet. Ein sortierte Ablage der Datensätze wird dadurch nicht erzwungen.

Ordnungsmerkmale sind dennoch genau wie einfache Merkmale Teil der statistischen Information bzw. der Daten. Aus fachlicher Sicht ist es egal, ob ein Merkmal als Ordnungsmerkmal oder als einfaches Merkmal übermittelt wird. Hauptzweck von Ordnungsmerkmalen ist die Redundanzverringerung, vor allem bei der Lieferung mittlerer und großer Datenmengen.

Innerhalb eines Datensegmentes müssen Ordnungsmerkmale stets gleich tief geschachtelt sein (d.h. alle Pfade vom höchsten bis zum niedrigsten Ordnungsmerkmal sind gleich lang). Ordnungsmerkmale der gleichen Hierarchiestufe müssen immer

den gleichen Namen haben, also dasselbe Ordnungskriterium beschreiben. Die Anwendung muss die Einhaltung dieser Einschränkung prüfen, da sie nicht über eine DTD bzw. ein Schema definierbar ist.

Regel: Enthält ein Datensegment Ordnungsmerkmale, so muss über alle Datensätze hinweg ihre Reihenfolge in der hierarchischen Anordnung unverändert bleiben.

2.5.2.3. Hilfsmerkmale

Hilfsmerkmale (`<hmm>`) sind solche Merkmale, die häufig zur Erhebung meldungsglobaler Zusatzinformationen verwendet werden; oft dienen sie auch zu organisatorischen und Verfahrenszwecken. Wie Ordnungsmerkmale haben sie einen Namen (Attribut `name`) und einen Wert (`<wert>`), der nicht leer sein darf. Hilfsmerkmale sind in den Knoten des Segmentpfades (`<nachricht>`, `<segment>`) erlaubt.

2.5.3. Merkmalsgruppen

Ein einfaches Merkmal darf in einem Datensatz grundsätzlich genau ein einziges Mal vorkommen. Es ist jedoch möglich, dass für ein Erhebungsobjekt mehrere Ausprägungen eines Merkmals existieren, die alle dem gleichen Erhebungskontext und -objekt zugeordnet sind (z.B. in einer zeitlichen Untergliederung wie Umsätze eines Betriebes nach Monaten). Oft handelt es sich dabei um mehrere Merkmale, die eine zusammenhängende, wiederholbare Struktur bilden (eine sogenannte *wiederholte Feldgruppe*). Dem steht die Eindeutigkeit von Merkmalen in einem Datensatz entgegen. Ein gangbarer Weg ist die Verteilung wiederholbarer Merkmale auf mehrere Datensätze, mit den Nachteilen größerer Redundanz (z.B. Wiederholung der Betriebsnummer je Datensatz) oder Komplexität (Betriebsnummer als Ordnungsmerkmal, n:1-Abbildung auf eine flache Satzstruktur).

Eine bessere Möglichkeit ist die Bildung von *Merkmalsgruppen* (`<mmgr>`). Merkmale müssen innerhalb einer Merkmalsgruppe eindeutig sein, aber in einem Datensatz kann es beliebig viele Instanzen einer Merkmalsgruppe geben. Merkmalsgruppen dürfen wiederum Merkmalsgruppen enthalten (s. 2.5.3.2).

2.5.3.1. Identifikation und Indizierung

Eine Merkmalsgruppe wird über den Namen (Attribut `name`) identifiziert. Der Zugriff auf eine Instanz einer Merkmalsgruppe erfolgt mittels eines numerischen ganzzahligen Indexes ≥ 1 , dessen Gültigkeitsbereich das übergeordnete Element `<satz>` bzw. `<mmgr>` ist. Der Index kann auf verschiedene Weise ermittelt oder angegeben werden; die Auswahl ist abhängig vom Attribut `index` und ggf. externen Beschreibungsinformationen.

DatML/RAW unterstützt folgende Methoden der Indizierung:

- *Direkte* Indizierung bedeutet, dass der numerische Index der Merkmalsgruppe explizit angegeben ist, z.B. `index="1"`;

- *Indirekte* Indizierung heißt, dass der numerische Index anhand eines Merkmals der Merkmalsgruppe gebildet wird, dessen Name mit dem Attribut `index` i.d.F. `index="name(merkmalsname)"` spezifiziert wird.
- *Automatisch-indirekte* Indizierung entspricht indirekter Indizierung; der Name des indexbildenden Merkmals entstammt aber nicht dem Attribut `index`, sondern einer externen Information, z.B. einer Datensatzbeschreibung.
- *Automatisch-direkte* Indizierung findet statt, wenn weder das Attribut `index` gesetzt noch eine externe Information zur Indexbildung vorhanden ist; der Index wird, beginnend mit 1, für jedes Vorkommen einer Merkmalsgruppe in der Reihenfolge der Ablage um 1 erhöht.

Regel:

- Merkmalsgruppen werden nicht über Datensätze hinweg indiziert, d.h. jeder Datensatz bildet einen separaten Werteraum für die Indizes der in ihm enthaltenen Merkmalsgruppen, und mit jedem Datensatz beginnt die Indizierung für die in ihm enthaltenen Merkmalsgruppen von vorne.
- Die Indizierungsmethoden werden in der Reihenfolge direkt, indirekt, automatisch-indirekt und automatisch-direkt angewandt.
- Die automatischen Indizierungsmethoden unterscheiden sich nur durch das Fehlen bzw. Vorhandensein externer Informationen über das indexbildende Merkmal, sie sind hinsichtlich des DatML/RAW-Dokumentes identisch.
- Innerhalb eines Datensegmentes müssen alle Instanzen einer Merkmalsgruppe die gleiche Indizierungsmethode benutzen, also den gleichen Wert für das Attribut `index` haben.
- Für jede Instanz einer Merkmalsgruppe muss ein eindeutiger Indexwert ermittelt werden können.
- Die Instanzen einer Merkmalsgruppe müssen weder nach dem Index sortiert sein noch ununterbrochenen aufeinander folgen.

Eine DatML/RAW-Anwendung sollte darüber hinaus festlegen:

- Welche Indizierungsmethoden sind für die Erhebung zulässig?
- Wie werden Konflikte zwischen der verwendeten Indizierungsmethode und externen Beschreibungsinformationen behandelt (z.B. abweichende Informationen über das indexbildende Merkmal)?
- Sind in einem Datensatz in den Indexwerten einer Merkmalsgruppe Lücken erlaubt, und wenn ja, wie werden diese behandelt (nicht-existent, Anfangswerte)?

Beispiele:

Direkter Index (gleiches Ergebnis):

```
<mmgr name="intensivbetten" index="1">
  <mm name="abrechnungsmonat"><wert>03</wert></mm>
  <mm name="insgesamt">...</mm>
```

```

    <mm name="belegungstage">...</mm>
    <mm name="patienten">...</mm>
  </mmgr>
  <mmgr name="intensivbetten" index="2">
    <mm name="abrechnungsmonat"><wert>05</wert></mm>
    <mm name="insgesamt">...</mm>
    <mm name="belegungstage">...</mm>
    <mm name="patienten">...</mm>
  </mmgr>

```

Indirekter Index; das indexbildende Merkmal ist `abrechnungsmonat`; die Indizierung weicht in diesem Fall (index=3,5) vom vorhergehenden Beispiel (index=1,2) ab:

```

  <mmgr name="intensivbetten" index="name(abrechnungsmonat)">
    <mm name="abrechnungsmonat"><wert>03</wert></mm>
    <mm name="insgesamt">...</mm>
    <mm name="belegungstage">...</mm>
    <mm name="patienten">...</mm>
  </mmgr>
  <mmgr name="intensivbetten" index="name(abrechnungsmonat)">
    <mm name="abrechnungsmonat"><wert>05</wert></mm>
    <mm name="insgesamt">...</mm>
    <mm name="belegungstage">...</mm>
    <mm name="patienten">...</mm>
  </mmgr>

```

Automatisch-direkter bzw. *automatisch-indirekter* Index, also Indizierung durch die Anwendung mit oder ohne Rückgriff auf externe Informationen.

```

  <mmgr name="intensivbetten">
    <mm name="insgesamt">...</mm>
    <mm name="belegungstage">...</mm>
    <mm name="patienten">...</mm>
  </mmgr>
  <mmgr name="intensivbetten">
    <mm name="insgesamt">...</mm>
    <mm name="belegungstage">...</mm>
    <mm name="patienten">...</mm>
  </mmgr>

```

HINWEIS: Diese Spezifikation regelt *nicht* die Behandlung *nicht-numerischer* Indizes. Es ist aber möglich, dass eine Anwendung mit Hilfe einer Umschlüsselungstabelle nicht-numerische Indizes auf numerische abbildet, wie hier für `abrechnungsmonat`:

```

  <mmgr name="intensivbetten" index="name(abrechnungsmonat)">
    <mm name="abrechnungsmonat">
      <wert>maerz</wert> <!-- ergibt 3 -->
    </mm>
    <mm name="insgesamt">...</mm>
    <mm name="belegungstage">...</mm>
    <mm name="patienten">...</mm>
  </mmgr>
  <mmgr name="intensivbetten" index="name(abrechnungsmonat)">
    <mm name="abrechnungsmonat">
      <wert>mai</wert> <!-- ergibt 5 -->
    </mm>
  </mmgr>

```

```
</mm>
<mm name="insgesamt">...</mm>
<mm name="belegungstage">...</mm>
<mm name="patienten">...</mm>
</mmgr>
```

2.5.3.2. Geschachtelte Merkmalsgruppen

Merkmalsgruppen können geschachtelt werden. Damit ist es möglich, innerhalb eines Datensatzes Strukturen mit beliebig tief angeordneten *Merkmalsebenen* zu beschreiben. Für geschachtelte Merkmalsgruppen gilt folgende zusätzliche Regel:

Regel:

- Merkmalsgruppen werden nicht über übergeordnete Merkmalsgruppen hinweg indiziert, d.h. jede Merkmalsgruppe bildet einen separaten Werteraum für die Indizes der in ihnen enthaltenen Merkmalsgruppen, und mit jeder Merkmalsgruppe beginnt die Indizierung für die in ihr enthaltenen Merkmalsgruppen von vorne.

2.5.4. Namen

2.5.4.1. Wertebereich

Ordnungsmerkmale, Hilfsmerkmale, einfache Merkmale und Merkmalsgruppen werden durch per Attribut `name` zugewiesene Namen identifiziert. Die Menge der technisch möglichen Namen unterliegt lediglich den Beschränkungen der XML-Spezifikation für Attributwerte und wird von dieser Spezifikation nicht weiter begrenzt. Aus der Anwendungssicht ist ggf. eine weitergehende Beschränkung sinnvoll, z.B. um bestimmte Zeichen auszuschließen oder anwendungsspezifischen oder organisatorischen Namenskonventionen zu genügen.

Regel: Einschränkungen des Wertebereiches von Namen sind anwendungsspezifisch.

2.5.4.2. Namenskonflikte

Innerhalb einer Meldung (s. 2.3.5) haben alle Merkmale ungeachtet ihres Typs sowie alle Merkmalsgruppen einen gemeinsamen Namensraum.

Regel: Namensgleichheit zwischen Merkmalen liegt vor, wenn das Attribut `name` ohne Berücksichtigung des Attributes `klasse` bei mehr als einem Merkmal den gleichen Wert hat.

Der Bereich eines Dokumentes, innerhalb dessen Merkmale auf Namensgleichheit geprüft werden, ist abhängig vom Merkmalstyp:

Regel: Hilfsmerkmale müssen innerhalb einer Meldung eindeutig sein, Ordnungsmerkmale sowie Wertmerkmale, die nicht Teil einer Merkmalsgruppe sind, inner-

halb eines Datensatzes.

Für Wertmerkmale in Merkmalsgruppen gilt:

Regel: Ein einfaches Merkmal darf in einem Datensatz nur dann mehr als einmal vorkommen, wenn es Teil einer Merkmalsgruppe ist; es darf innerhalb einer Merkmalsgruppe nur einmal vorkommen.

Der Vergleich der Werte sollte nach der Normalisierung von Leerzeichen und der Auflösung von Character Entities (z.B. 'ä', ' ') erfolgen; letzteres erfolgt i.d.R. durch den XML-Parser. Diese Spezifikation legt nicht fest, ob ein Namenskonflikt nur den Datensatz, das Datensegment, die Nachricht oder das gesamte Dokument ungültig macht.

2.5.5. Sortierung

DatML erfordert keinerlei Sortierung und betrachtet Ordnungsmerkmale, Datensätze Merkmalsgruppen und Merkmale in keinem Fall als sortiert. Ordnungsmerkmale bewirken lediglich die Zuordnung von nachgeordneten Datensätzen bzw. Ordnungsmerkmalen zu genau einer Ausprägung auf der aktuellen Hierarchiestufe. Die Folge der Ausprägungen eines Ordnungsmerkmals muss hingegen nicht sortiert sein (s. 2.5.2.2).

2.5.6. Numerische Werte

Die Werte statistischer Merkmale werden im Element `<wert>` als Bestandteil von einfachen Merkmalen oder Ordnungsmerkmalen abgelegt. Für numerische Werte gelten dabei folgende Einschränkungen:

Regel: Numerische Werte statistischer Merkmale haben folgendes Format (Angaben in [] sind optional):

- Das allgemeine Format ist `[w][s]a[e[s]n][w]`:
 - `w` : Leerraum in ASCII-Kodierung gemäß XML-Spezifikation, also Leerzeichen (X'20'), Tabulatorzeichen (X'09'), Wagenrücklauf (X'0D') und Zeilenvorschub (X'0A'); für DatML/RAW-Dokumente in EBCDIC-Kodierung ist als Leerraum das Leerzeichen (X'40') zulässig, die Kodierungen der übrigen Zeichen können anwendungsspezifisch sein.
 - `s` : Vorzeichen („+“, „-“)
 - `a` : Betrag (0-9, optional Tausender-Trennzeichen und Dezimalzeichen)
 - `e` : Exponentenkennzeichen („e“, „E“)
 - `n` : Exponent (eine natürliche Zahl)
- Erlaubter Leerraum am Anfang und am Ende eines Wertes wird ignoriert.
- Insignifikante (führende und folgende) Nullen sind erlaubt und werden ignoriert.
- Das Dezimalzeichen per Default oder Definition mit dem Elementtyp + Attribut `<datenattribute dezimalzeichen="..."/>`.
- Das Tausender-Trennzeichen per Default oder Definition mit dem Elementtyp +

Attribut `<datenattribute tausender-trennzeichen="..." />`.

- Dezimalzeichen und Tausender-Trennzeichen müssen verschieden sein.
- Das erste von Leerraum verschiedene Zeichen muss ein Vorzeichen oder das erste Zeichen des Betrages sein.
- Ein Betrag beginnt mit einer Ziffer; enthält er weitere Zeichen, muss es sich um weitere Ziffern, Tausender-Trennzeichen bzw. Dezimalzeichen handeln. Einem Tausender-Trennzeichen müssen genau drei Ziffern folgen; folgt danach ein weiteres Zeichen, muss es ein zulässiges, von einer Ziffer verschiedenes Zeichen sein, das den Betrag fortsetzt oder beendet. Folgt nach dem Dezimalzeichen eine Ziffer, dürfen bis zum Ende des Betrages nur noch Ziffern folgen. Folgt auf das Dezimalzeichen keine Ziffer, wird das Dezimalzeichen ignoriert (keine Nachkommastellen). Der Betrag endet mit dem Ende der Zeichenkette oder mit dem Exponentenkennzeichen oder einem Leerraumzeichen; endet er mit einem Leerraumzeichen, müssen alle evtl. folgenden Zeichen ebenfalls Leerraumzeichen sein.
- Nach dem Exponentenkennzeichen darf nur ein optionales Vorzeichen gefolgt vom obligatorischen Exponentenbetrag gefolgt von optionalem Leerraum vorkommen. Der Exponentenbetrag darf nur aus Ziffern bestehen.

Beispiele:

```
001
+12
-37,80
0,657
2,46E-6
1.312.444,2
```

Die Attribute `dezimalzeichen` und `tausender-trennzeichen` stehen über den Elementtyp `<datenattribute>` nachrichten- bzw. segmentbezogen zur Verfügung.

Die Behandlung numerischer Werte von Metadaten wird durch diese Spezifikation nicht geregelt.

2.5.7. 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.

2.5.7.1. Klasse `datum`

format	
Bedeutung:	Ein Einzeldatum oder eine Zeitspanne aus zwei Einzeldaten. S. 3.4.2.
Formatschema:	<p><code>datum[-datum]</code></p> <p><code>datum: j4[{{h hh}}]{{q qq}}[s ss]{{w ww}}monat[tag[std[<i>min</i>[<i>sek</i>]]]]</code></p> <p><code>j4</code> : <code>jjjj</code>, Jahr vierstellig</p> <p><code>h</code> : <code>h</code>, Halbjahr i.d.F. <code>hn Hn n</code>, <code>n=1..2</code></p> <p><code>hh</code> : <code>hh</code>, Halbjahr i.d.F. <code>nn</code>, <code>nn=01..02</code></p> <p><code>q</code> : <code>q</code>, Quartal i.d.F. <code>qn Qn n</code>, <code>n=1..4</code></p> <p><code>qq</code> : <code>qq</code>, Quartal i.d.F. <code>nn</code>, <code>nn=01..04</code></p> <p><code>s</code> : <code>s</code>, Semester i.d.F. <code>sn Sn n</code>, <code>n=1..2</code>¹⁾</p> <p><code>ss</code> : <code>ss</code>, Semester i.d.F. <code>nn</code>, <code>nn=01..02</code>¹⁾</p> <p><code>monat: mm</code>, Monat zweistellig</p> <p><code>w</code> : <code>w</code>, Woche i.d.F. <code>wnn Wnn nn</code>, <code>nn=01..53</code></p> <p><code>ww</code> : <code>ww</code>, Woche i.d.F. <code>nn</code>, <code>nn=01..53</code></p> <p><code>tag</code> : <code>tt</code>, Tag zweistellig</p> <p><code>std</code> : <code>hh</code>, Stunde zweistellig</p> <p><code>min</code> : <code>mm</code>, Minute zweistellig</p> <p><code>sek</code> : <code>ss</code>, Sekunde zweistellig</p>
	<p>¹⁾ Das Sommersemester ist mit dem Wert 1 und das Wintersemester mit dem Wert 2 zu kodieren.</p>
Hinweise:	<ul style="list-style-type: none"> • Das Jahr ist stets vierstellig und obligatorisch. • Auf das Jahr folgt entweder ein Monat, ein Quartal, ein Halbjahr oder eine Woche (von 01-53). • Die Kennzeichnung eines Halbjahres, eines Semesters, eines Quartals und einer Woche im <i>Datum</i> (<i>nicht</i> im Format) durch einen Buchstaben ist optional; so kann das erste Halbjahr 2002 wahlweise als <code>2002H1</code> und als <code>20021</code> angegeben werden. • Ein Tag darf nur auf einen Monat folgen. • Eine Uhrzeit darf nur auf einen Tag folgen.

Beispiele:

Formatschema	Werte	Bedeutung
jjjjmm	200205	Jahr und Monat
jjjj-jjjj	2001-2002	Jahresspanne
jjjjq	2002Q2, 20022	Jahr und zweites Quartal
jjjjqq	200202	Jahr und zweites Quartal
jjjjh	2002H1, 20021	Jahr und erstes Halbjahr
jjjjhh	200201	Jahr und erstes Halbjahr
jjjjs	20021, 2002S1	Jahr und Sommersemester
jjjjss	200202	Jahr und Wintersemester

2.6. Verarbeitung *[nicht-normativ]*

Wohlgeformte DatML/RAW-Dokumente können wie jedes andere XML-Dokument mit zur Verfügung stehenden Technologien wie Parsern, XSLT-Engines usw. verarbeitet werden.

2.6.1. Verteilung

Das Datenmodell von DatML/RAW erlaubt die gleichzeitige Lieferung von Daten in beliebiger Kombination von Erhebung, Berichtszeitraum, Berichtspflichtigem etc. und ist zudem mandantenfähig. Die Kehrseite dieser Flexibilität ist die Anforderung, DatML/RAW-Dokumente in kleinere Einheiten zerlegen und diese verteilen zu können.

Eine Verteilung ist nach zwei Kriterien möglich und findet ggf. in verschiedenen Stufen des Bearbeitungsprozesses statt (s. 2.2.2.2).

- *Empfängerorientierte Verteilung:* Ein DatML/RAW-Dokument wird in Dokumente zerlegt, die jeweils alle für einen Berichtsempfänger bestimmten Daten enthalten.
- *Kontextorientierte Verteilung:* Ein DatML/RAW-Dokument wird in Dokumente zerlegt, die jeweils alle Daten für einen bestimmten statistischen Kontext – unter Ausschluss des Berichtsempfängers – enthalten.

HINWEIS: Der Berichtsempfänger ist zwar streng genommen Teil des statistischen Kontextes (s. 2.3.4), wird aber mit Blick auf die Mandantenfähigkeit von DatML/RAW getrennt behandelt.

Es empfiehlt sich, grundsätzlich zuerst eine empfänger- und dann eine kontextorientierte Verteilung vorzunehmen. Falls notwendig, wird ein Zwischendokument erzeugt, z.B. wenn zwischen den beiden Verteilungsschritten eine Datenübertragung erfolgt. Ein Dokument kann aber auch in einem Schritt nach beiden Kriterien zerlegt und verteilt werden.

2.6.2. Zerlegung

Die ursprüngliche Segmentierung eines Dokuments muss bei einer Zerlegung nicht notwendigerweise erhalten bleiben. Es kann z.B. sinnvoll sein, eine hierarchische durch eine flache (dafür hinsichtlich der Metadaten redundante) Segmentierung oder eine entsprechende Anzahl von Nachrichten zu ersetzen, wenn dies nachfolgende Prozesse vereinfacht.

Folgendes muss bei einer Zerlegung beachtet werden:

- Der neue Absender des Dokumentes ist entweder der ursprüngliche Absender oder die Stelle, die das Teildokument erzeugt hat.
- Der neue Empfänger ist bei empfängerorientierter Verteilung der Berichtsempfänger, bei kontextorientierter Verteilung ist es der ursprüngliche Empfänger.

- Gleichheit des Empfängers liegt vor, wenn alle zu dessen Identifikation herangezogenen Eigenschaften (Elemente und Attribute) in ihren Werten gleich sind. Die Auswahl dieser Eigenschaften und Verarbeitungsdetails (z.B. Vergleich von Attributwerten und Elementinhalt mit oder ohne Leerzeichen) sind anwendungsabhängig.
- Gleichheit des Kontextes liegt vor, wenn alle dessen Eigenschaften (Elemente und Attribute) mit Ausnahme des Berichtsempfängers in ihren Werten gleich sind.
- Jedes Datensegment kann in ein separates Top-Level-Segment oder eine separate Nachricht ausgegeben werden, jeweils mit einem vollständigen Metadatenatz.
- Die Verarbeitung des Eingabedokumentes erfolgt entlang der Segmentpfade. Der Segmentbaum sollte in Vorordnung traversiert werden, da dies der Elementfolge entspricht, d.h. bei SAX-Parsing wird keine Zwischenpufferung notwendig.
- Beim Traversieren eines Segmentpfades werden die auf ihm eingetragenen Metadaten „eingesammelt“, bis das Blatt, also das Datensegment erreicht ist (Achtung: Das Datensegment, das vollständig in einem Element `<datensegment>` enthalten ist, liegt im letzten Segmentknoten des Segmentpfades, und auch hier können noch Metadaten vorhanden sein). Die Anwendung prüft bei diesem Schritt die Metadaten auf Redundanzfreiheit.
- Bei der Übernahme von Datensegmenten darf es keine Änderung an den vorhandenen Merkmalen und Werten geben. Um dies zu vermeiden, kann ein Datensegment Element für Element unverändert in die Ausgabe übernommen werden (sog. Identity Transformation). Falls erforderlich oder gewünscht, bieten sich eine Reihe von Prüf-, Bereinigungs- und Vorbereitungsmaßnahmen an:
 - Merkmale in den Datensätzen sortieren.
 - Ordnungsmerkmale in einfache Merkmale umwandeln.
 - Fehlende Merkmale bzw. Merkmalsgruppen ergänzen (Anfangswerte).
 - Prüfung von Vollständigkeit und Indexwerten.
 - Prüfung numerischer Werte.
- Jedes Ausgabedokument bleibt nach Verarbeitung eines Datensegmentes je nach gewünschter Ausgabestruktur auf das Ende des Segmentes mit dem Datensegment, des Segmentes oder der Nachricht positioniert, da das Eingabedokument weitere Daten für den selben Empfänger oder mit dem selben Kontext enthalten kann, und wird erst geschlossen, wenn das gesamte Eingabedokument verarbeitet worden ist.

2.7. Weiteres

2.7.1. Informationen aus Online-Erhebungen

Die in HTML-Fragebögen verwendeten Feldnamen können mit Hilfe des Attributes `html-name` für folgende Elementtypen dokumentiert werden:

- Alle in `<adresse>`, `<organisation>` und `<person>` erlaubten Elementtypen mit Dateninhalt (PCDATA)
- Merkmale und Merkmalsgruppen: `<mm>`, `<hmm>`, `<omm>`, `<mmgr>`
- Elementtypen `<telefon>`, `<email>`, `<fax>` und `<url>`
- Elementtyp `<zeile>`

2.7.2. Unterstützung zeilenbasierter Formate

Erhebungsverfahren verwenden häufig zeilenbasierte Datenmodelle für Identifikations- und Kontaktangaben. Für die Abbildung solcher Datenmodelle bietet DatML/RAW zwei Alternativen: die *zeilenbasierte Ablage* und die *Abbildung typbasierter auf zeilenbasierte Einträge*.

HINWEIS: Es wird empfohlen, typbasierte Datenmodelle und die entsprechenden DatML/RAW-Strukturen zu benutzen. Zeilenbasierte Datenmodelle werden nur aus unterstützt, um die Integration von DatML/RAW in existierende Erhebungen zu erleichtern.

2.7.2.1. Zeilenbasierte Ablage

Die alternative zeilenweise Ablage per Elementtyp `<zeile>` ist in den Elementtypen `<identifikation>` und `<kontakt>` möglich. Das Attribut `zeilennummer` erlaubt die direkte Angabe von Zeilennummern, nicht notwendige Zeilen können so z.B. bei Korrekturen entfallen. Details der Verwendung und Sicherstellung der Konsistenz ist Sache der Anwendung.

```
<identifikation>
  <zeile zeilennummer="1">Martin Mustermann</zeile>
  <zeile zeilennummer="2">Hauptstraße 99</zeile>
  <zeile zeilennummer="3">99999 Musterstadt</zeile>
</identifikation>
<kontakt>
  <zeile zeilennummer="1">09999/9999</zeile>
</kontakt>
<korrektur>
  <identifikation>
    <zeile zeilennummer="2">Bahnhofstraße 11</zeile>
  </identifikation>
</korrektur>
```

2.7.2.2. Abbildung von Typen auf Zeilen

Die explizite Abbildung von typbasierten auf zeilenbasierte Informationen erfolgt mit Hilfe des Attributes `zeilennummer`:

```
<identitaet>
  <person>
    <vorname zeilennummer="1">Mustermann</vorname>
    <nachname zeilennummer="1">Martin</nachname>
  </person>
</identitaet>
<adresse>
  <strasse zeilennummer="2">Hauptstraße 99</strasse>
  <plz zeilennummer="3">99999</plz>
  <ort zeilennummer="3">Musterstadt</ort>
</adresse>
<kontakt>
  <telefon zeilennummer="9">09999/9999</telefon>
</kontakt>
<korrektur>
  <adresse>
    <strasse Bahnhofstraße 11</strasse>
  </adresse>
</korrektur>
```

Details der Verwendung und Sicherstellung der Konsistenz ist Sache der Anwendung. Es ist z.B. denkbar, typbasierte Einträge auch implizit auf Zeilen abzubilden, wenn das Zeilenmodell entsprechenden Beschränkungen unterliegt (z.B. wenn Zeile 1 immer den vollständigen Namen, Zeile 2 die Straße enthält, usw.).

Als Richtschnur wird folgendes Vorgehen *empfohlen*:

- Jeder Elementtyp wird *explizit* auf eine Zeile abgebildet.
- Wenn mehr als ein Elementtyp auf dieselbe Zeile abgebildet wird, werden die Elementinhalte – in der Reihenfolge ihres Auftretens oder in einer definierten Reihenfolge – zu einer Resultatzeichenfolge zusammengesetzt; falls notwendig, sollte ein zusätzlich eingefügtes Leerzeichen zwei Werte trennen.
- Für Zwecke des Abgleichs werden aus der Resultatzeichenfolge und einem vorhandenen Vergleichswert zuvor alle Leerzeichen entfernt oder normalisiert.
- Falls ein Verfahren vorschreibt, dass im Falle von Korrekturen die vorhandenen, zu korrigierenden Werte ebenfalls geliefert werden müssen, sollte die Abbildung anhand der vorhandenen Werte spezifiziert und ausgeführt werden. Die Korrekturen selbst sollten typbasiert sein und keine Abbildungsinformationen verwenden. Es können so keine Werte korrigiert werden, die nicht als vorhanden geliefert wurden, da für sie keine Abbildung existiert. Falls die Abbildung sowohl für die vorhandenen Werte als auch für die Korrekturen spezifiziert ist, muss sie für beide identisch sein.
- Für `zeilennummer` wird ein Wertebereich definiert.

2.7.3. Vorgeschriebene Elemente

Das Datenmodell von DatML/RAW enthält eine Reihe vorgeschriebener Elemente, die in einem gültigen Dokument nicht fehlen dürfen. Diese Elemente bilden ein Grundgerüst für jedes DatML/RAW-Dokument. In einigen Fällen, z.B. bei Adressangaben, ist es möglich, dass für vorgeschriebene Elemente keine (sinnvollen) Daten zur Verfügung stehen. Es ist dann zulässig, leere Elemente zu erzeugen. *Nicht* betroffen sind die eigentlichen statistischen Rohdaten, da diese in vollem Umfang erhebungsspezifisch sind und mit erhebungsneutralen Strukturen abgebildet werden, die lediglich in der Anzahl und ggf. Schachtelungstiefe variieren.

Es ist möglich, dass eine Erhebung bestimmte Angaben vorschreibt, die in optionalen Elementen abgelegt werden. In diesem Fall tritt zur Validierung durch eine DTD oder ein Schema die Validierung durch die Anwendung.

HINWEIS: Ein Element darf nicht leer gelassen werden, wenn es selbst nicht-optionale Elemente enthält. Leere Elemente sind daher nur möglich, wenn ihr Inhaltsmodell nur Daten und/oder ausschließlich optionale Elemente erlaubt.

2.7.4. Einfügen von Daten

Der Inhalt eines XML-Dokumentes besteht aus Daten und Markup. Beide Komponenten sind textbasiert und müssen immer eindeutig unterschieden werden können. Daten dürfen deshalb keine Zeichen enthalten, die das korrekte Erkennen des Markup erschweren oder verhindern.

Es wird daher empfohlen, keine ungeprüften Daten in ein DatML/RAW-Dokument zu übernehmen. Dies gilt besonders dort, wo dynamische Inhalte in einem DatML/RAW-Dokument abgelegt werden, z.B. bei der Übernahme von Online-Daten aus HTML-Fragebögen. Ggf. müssen Daten entsprechend aufbereitet werden, z.B. durch die Umwandlung von Zeichen in Character Entities. Die Definition von Markup und Mechanismen zur Vermeidung von Konflikten mit Dateninhalten sind in der [XML-Spezifikation des World Wide Web Consortiums](#) beschrieben.

3. Elementtypen

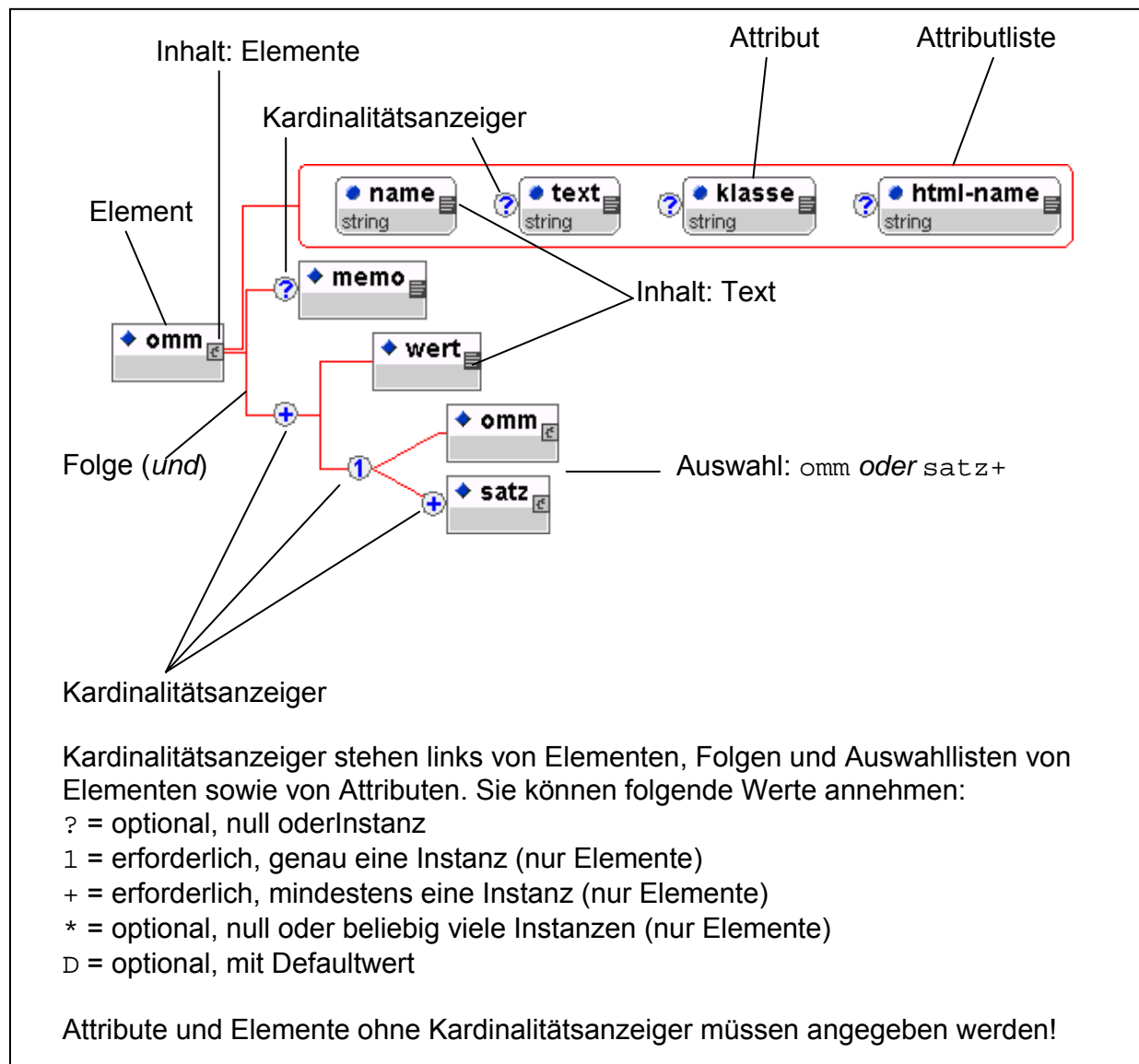
Elementtypen sind in der Beschreibung nach funktionalen und strukturellen Kriterien gruppiert und innerhalb jeder Gruppe alphabetisch aufsteigend sortiert; Attribute sind mit ihren Elementtypen beschrieben:

3.1 Darstellung der Elementtypen und Attribute

3.2 Dokument

3.1. Darstellung der Elementtypen und Attribute

3.1.1. Graphische Darstellung der Elementtypen



3.1.2. Tabellarische Darstellung der Elementtypen

Element	<i>elementname</i>		
Status	<i>status</i>	[<i>r</i>]	<i>version</i>
Datentyp:	<i>datentyp</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	<i>nsp</i>
<i>graphisches Inhaltsmodell</i> (s. 3.1.1)			

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.

nsp gibt – falls verwendet – das Namensraumpräfix an, das in einer DTD für den Elementtypen definiert ist.

3.1.3. Tabellarische Darstellung der Attribute

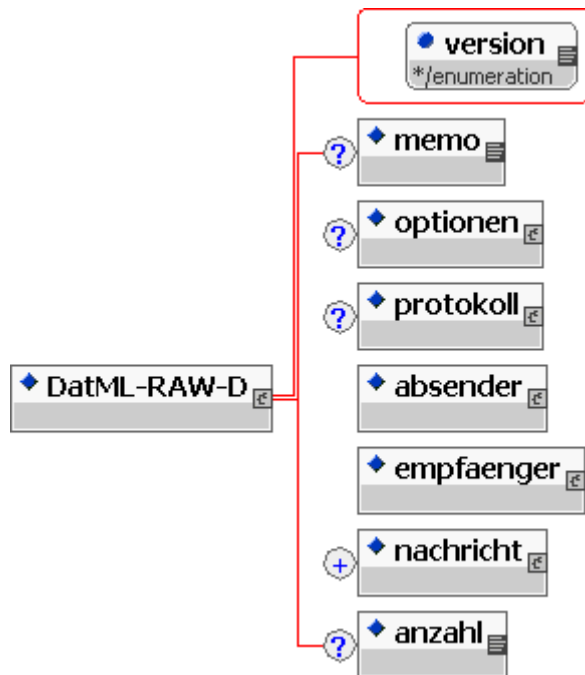
Die Attribute eines Elementtyps sind in einer eigenen Tabelle mit folgenden Spalten beschrieben:

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 Wertebereich des Attributes	
	Datentyp	Beschreibung
	<i>char</i> [<i>n..m</i>]	Zeichen [zulässige Länge]
	<i>int</i> [<i>n..m</i>]	Ganzzahlen [Wertebereich]
	<i>enumeration</i>	Eine Liste vordefinierter Werte

3.2. Dokument, Nachricht, Segment

3.2.1. Element <DatML-RAW-D>

Element	<DatML-RAW-D>	
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



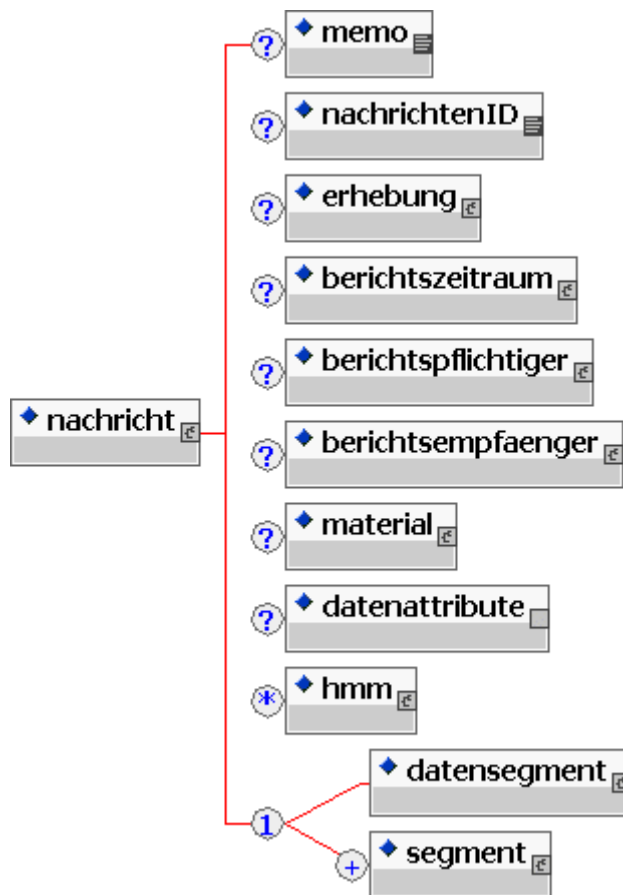
Attribute:	S Datentyp, <u>Default</u>, Werte
version	M <i>enumeration:2.0</i> ; Versionsbezeichnung

Details

Eine Datenlieferung enthält genau eine Absenderangabe, genau eine Empfängerangabe und mindestens eine "Rohdatennachricht". Optional sind ein Kommentar (memo), Verarbeitungsoptionen, Protokolldaten (Erstellung und ggf. Informationen über den Dateneingang), und zu Kontrollzwecken die Anzahl der Nachrichten. Der Teilname '-D' zeigt die deutsche Version des Dokumenttyps an.

3.2.2. Element <nachricht>

Element	<nachricht>	
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



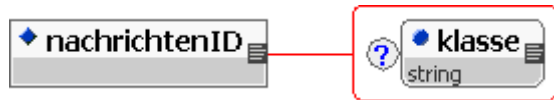
Details

Eine Nachricht besteht aus einem optionalen Kommentar und einer Auswahl optionaler Metadaten, gefolgt von entweder einem Datensegment (<datensegment>), welches die Meldungsdaten enthält, oder einer Folge von Segmenten (<segment>), die wiederum weitere Segmente enthalten können.

Die Reihenfolge der Metadaten ist beliebig, jedoch darf jedes Metadatum auf einem Segmentpfad nur einmal vorkommen, egal, ob es sich in einem Nachrichten- oder einem Segmentknoten befindet; s. 2.3.3. Hilfsmerkmale (<hmm>) dürfen beliebig oft erscheinen, doch muß jedes einen eindeutigen Namen haben.

3.2.3. Element <nachrichtenID>

Element	<nachrichtenID>	
Status:	neu	≥ 2.0
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



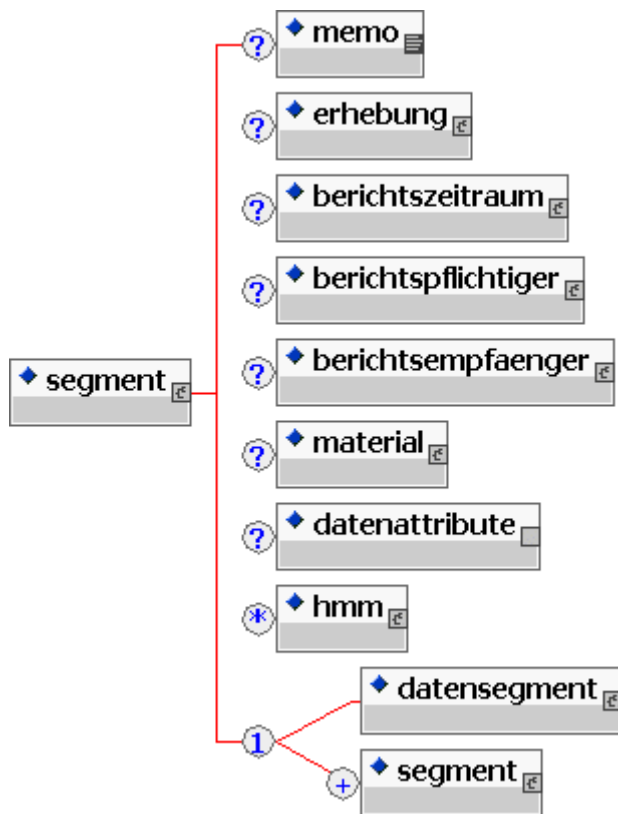
Attribute: **S** **Datentyp, Default, Werte**
 klasse ? *char*; Klasse oder Klassifikation

Details

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

3.2.4. Element <segment>

Element	<segment>	
Datentyp:	element-content	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



Details

Ein Segment besteht aus einem optionalen Kommentar und mindestens einem Metadatum, gefolgt von einem Datensegment (<datensegment>), welches die Meldungsdaten enthält, oder einer Folge von Segmenten (<segment>), die wiederum weitere Segmente enthalten können.

Wie in einem Nachrichtenknoten (<nachricht>) ist die Reihenfolge der Metadaten beliebig, jedoch darf jedes Metadatum auf einem Segmentpfad nur einmal vorkommen, egal, ob es sich in einem Nachrichten- oder einem Segmentknoten befindet; s. 2.3.3. Hilfsmerkmale (<hmm>) dürfen beliebig oft erscheinen, doch muß jedes einen eindeutigen Namen haben.

Segmente können hierarchisch angeordnet sein, da das Element <segment> eine Rekursion auf sich selbst erlaubt. Ein Datensegmentknoten (<datensegment>) beendet die Rekursion. Die Menge der effektiven Metadaten bezogen auf eine Datenmenge ist die Gesamtmenge der Metadaten in einem Segmentpfad, d.h. vom Nachrichtenknoten bis zu dem Segmentknoten, der das Datensegment enthält, sowie den meldungsspezifischen Metadaten im Datensegment selbst..

Beispiele:

```
<!-- Eine Erhebung, 2 Berichtszeitraeume -->

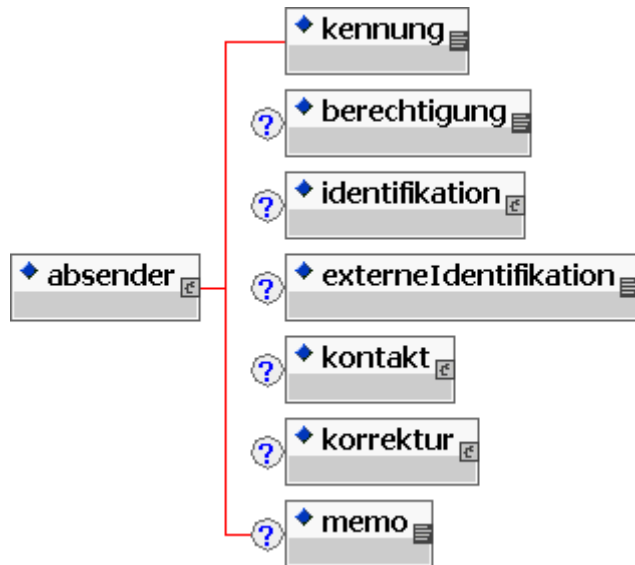
<segment>
  <erhebung>
    <kennung klasse="EVAS">51111</kennung>
  </erhebung>
  <segment>
    <berichtszeitraum>
      <jahr>2001</jahr>
      <monat>01</monat>
    </berichtszeitraum>
    <datensegment>
      <satz> ... </satz>
    </datensegment>
  </segment>
  <segment>
    <berichtszeitraum>
      <jahr>2001</jahr>
      <monat>02</monat>
    </berichtszeitraum>
    <datensegment>
      <satz> ... </satz>
    </datensegment>
  </segment>
</segment>

<!-- Ein Berichtszeitraum, 2 Erhebungen
      Metadaten können auch unter <nachricht> hängen!
-->
<nachricht>
  <berichtszeitraum>
    <jahr>2000</jahr>
    <monat>01</monat>
  </berichtszeitraum>
  <segment>
    <erhebung>
      <kennung klasse="EVAS">51111</kennung>
      <text>Einfuhr</text>
    </erhebung>
    <datensegment>
      <satz> ... </satz>
    </datensegment>
  </segment>
  <segment>
    <erhebung>
      <kennung klasse="EVAS">51121</kennung>
      <text>Ausfuhr</text>
    </erhebung>
    <datensegment>
      <satz> ... </satz>
    </datensegment>
  </segment>
</nachricht>
```

3.3. Verfahrensteilnehmer, Identifikation, Kontaktangaben

3.3.1. Element <absender>

Element	<absender>	
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

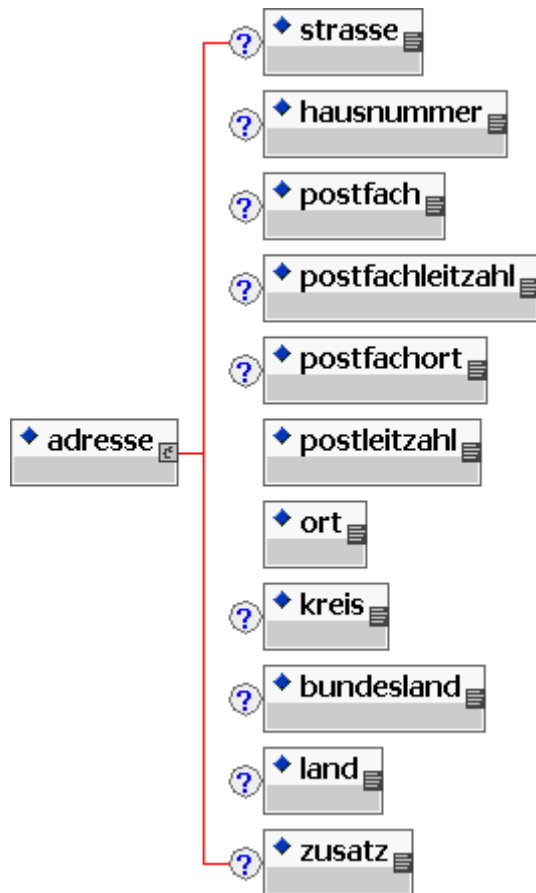


Details

Es ist anwendungsspezifisch festzuliegen, welche Daten zu Verifikation der Identität des Absenders herangezogen werden.

3.3.2. Element <adresse>

Element	<adresse>	
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



Details

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

3.3.3. Element <anrede>

Element	<anrede>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



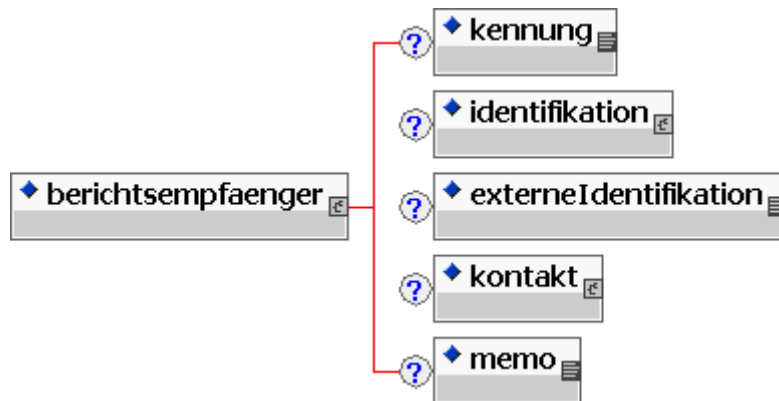
Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilennummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

Die Anrede einer Person.

3.3.4. Element <berichtsempfaenger>

Element	<berichtsempfaenger>	
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

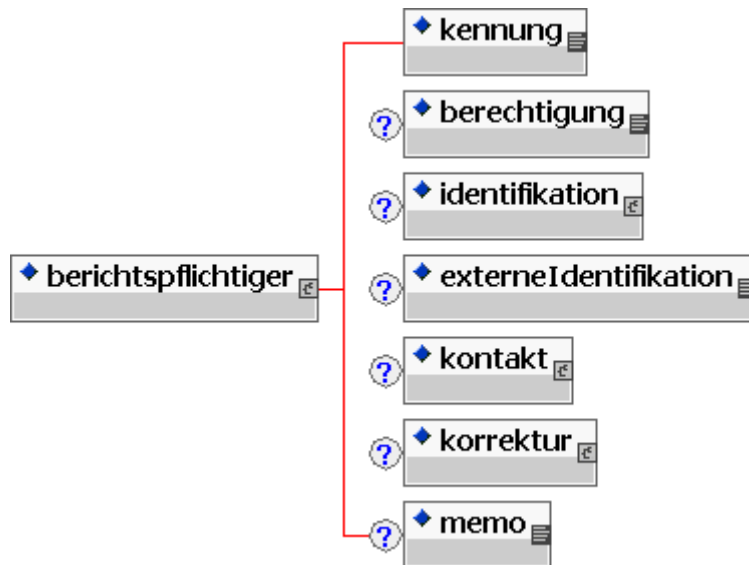


Details

Berichtsempfänger ist i.d.R. ein Statistisches Landesamt oder das Statistische Bundesamt. Der Berichtsempfänger muss in der Lage sein, sich selbst als Adressaten zu bestätigen und benötigt deshalb eine Angabe zu seiner Identifikation. Standardmäßig ist der Berichtsempfänger mit dem Empfänger des Dokumentes identisch.

3.3.5. Element <berichtspflichtiger>

Element	<berichtspflichtiger>	
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

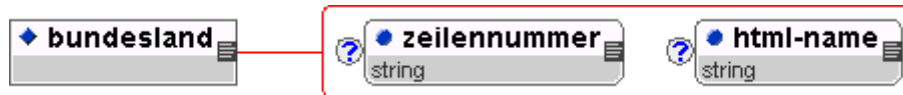


Details

Es ist anwendungsspezifisch festzuliegen, welche Daten zu Verifikation der Identität des Berichtspflichtigen herangezogen werden.

3.3.6. Element <bundesland>

Element	<bundesland>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



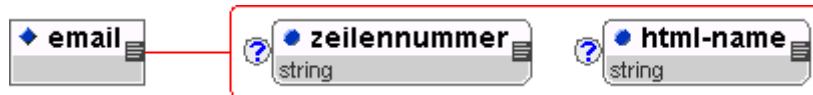
Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilennummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

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

3.3.7. Element <email>

Element	<email>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



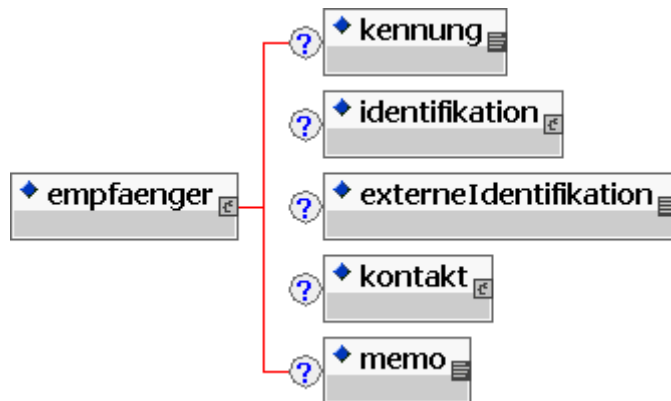
Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilennummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

Eine E-Mail-Adresse.

3.3.8. Element <empfaenger>

Element	<empfaenger>	
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



Details

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

3.3.9. Element <externeIdentifikation>

Element	<externeIdentifikation>	
Status:	neu	≥ 2.0
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



Attribute: **S** **Datentyp, Default, Werte**
klasse **D** *char*, Optionale Klasse oder Klassifikation

Details

Eine externe (i.d.R. vom Absender verwendete) Angabe für die Identifikation eines Verfahrensbeteiligten.

3.3.10. Element <fax>

Element	<fax>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilennummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

Eine Fax-Nummer.

3.3.11. Element <hausnummer>

Element	<hausnummer>	
Status:	neu	≥ 2.0
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



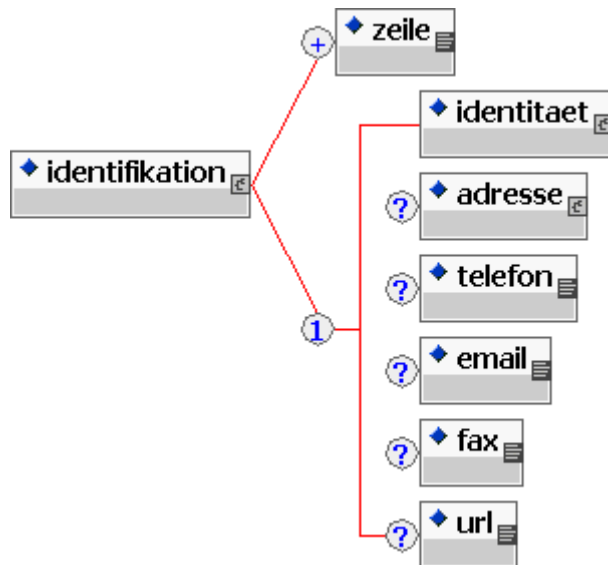
Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilennummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

Eine Hausnummer.

3.3.12. Element <identifikation>

Element	<identifikation>	
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

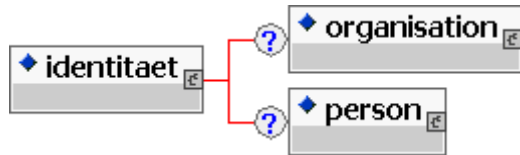


Details

<identifikation> erlaubt wahlweise zeilenbasierte oder strukturierte Angaben.

3.3.13. Element <identitaet>

Element	<identitaet>		
Datentyp:	<i>element-content</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-

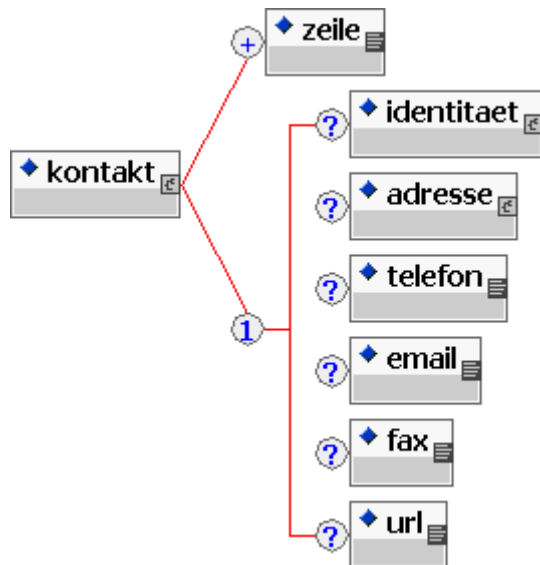


Details

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

3.3.14. Element <kontakt>

Element	<kontakt>	
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

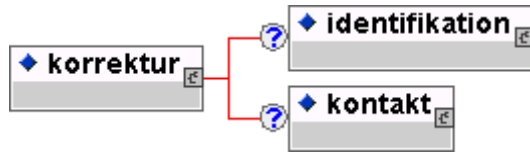


Details

<kontakt> erlaubt wahlweise zeilenbasierte oder strukturierte Angaben.

3.3.15. Element <korrektur>

Element	<korrektur>		
Datentyp:	<i>element-content</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-

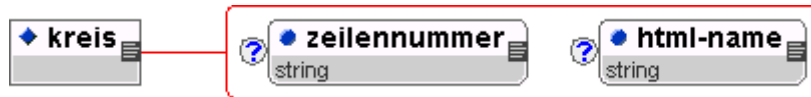


Details

<korrektur> erlaubt die Übermittlung von Korrekturdaten für die Identifikation und Kontaktinformationen des Absender und der Berichtspflichtigen.

3.3.16. Element <kreis>

Element	<kreis>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilennummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

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.

3.3.17. Element <land>

Element	<land>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



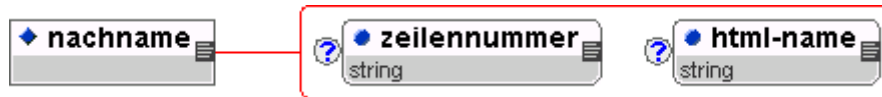
Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilennummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

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

3.3.18. Element <nachname>**Element** <nachname>Datentyp: *char*Namespace: <http://www.destatis.de/schema/datml-raw/2.0/de>

DTD: -

**Attribute:**

zeilennummer

S Datentyp, Default, Werte? *int*, Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,

html-name

? *char*, Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.**Details**

Der oder die Nachnamen einer natürlichen Person.

3.3.19. Element <niederlassung>

Element	<niederlassung>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



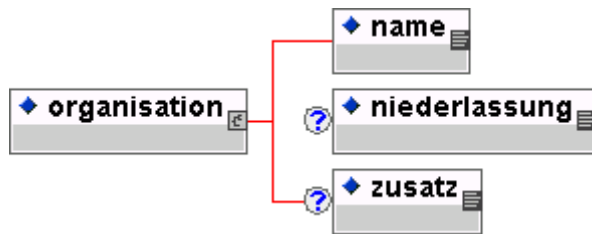
Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilennummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

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

3.3.20. Element <organisation>

Element	<organisation>	
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



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).

3.3.21. Element <ort>

Element	<ort>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



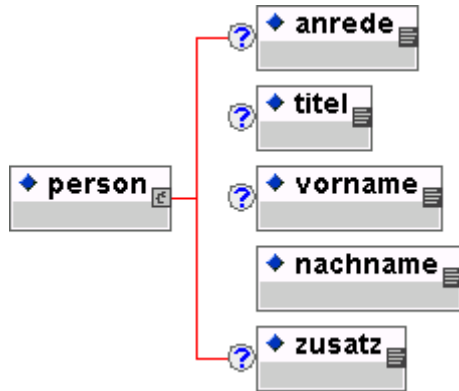
Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilenummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

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

3.3.22. Element <person>

Element	<person>	
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

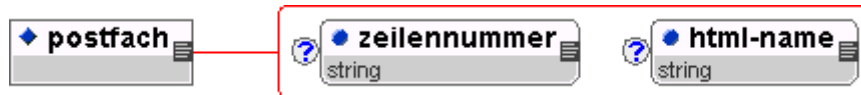


Details

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

3.3.23. Element <postfach>

Element	<postfach>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



Attribute:	S	Datentyp, <u>Default</u>, Werte
zeilennummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

Eine Postfachangabe.

3.3.24. Element <postfachleitzahl>

Element	<postfachleitzahl>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



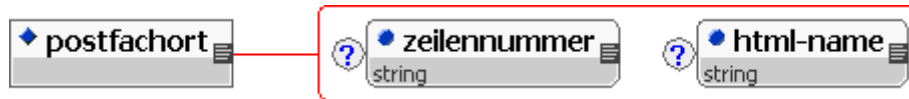
Attribute:	S	Datentyp, <u>Default</u>, Werte
zeilennummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

Die Leitzahl eines Postfachs.

3.3.25. Element <postfachort>

Element	<postfachort>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilennummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

Der Ort eines Postfachs.

3.3.26. Element <postleitzahl>

Element	<postleitzahl>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



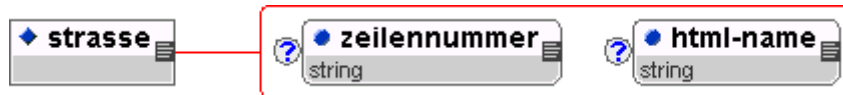
Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilennummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

Eine Postleitzahl.

3.3.27. Element <strasse>

Element	<strasse>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



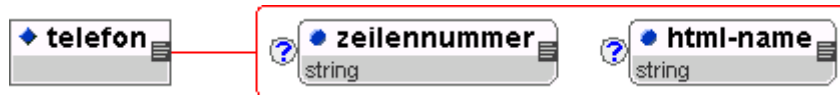
Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilennummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

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

3.3.28. Element <telefon>

Element	<telefon>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilenummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

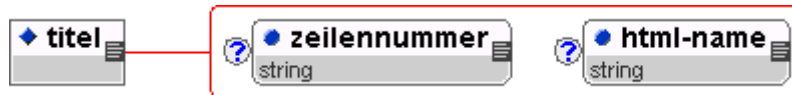
Eine beliebige Telefonnummer.

3.3.29. Element <titel>

Element <titel>

Datentyp: *char*

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



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

zeilennummer ? *int*, Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,

html-name ? *char*, Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

Adelstitel, akademische Titel oder sonstige Titel einer natürlichen Person.

3.3.30. Element <url>

Element	<url>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilennummer	?	<i>int</i> ; Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> ; Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

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

3.3.31. Element <vorname>

Element	<vorname>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilennummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

Der oder die Vornamen einer natürlichen Person.

3.3.32. Element <zeile>

Element	<zeile>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilennummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

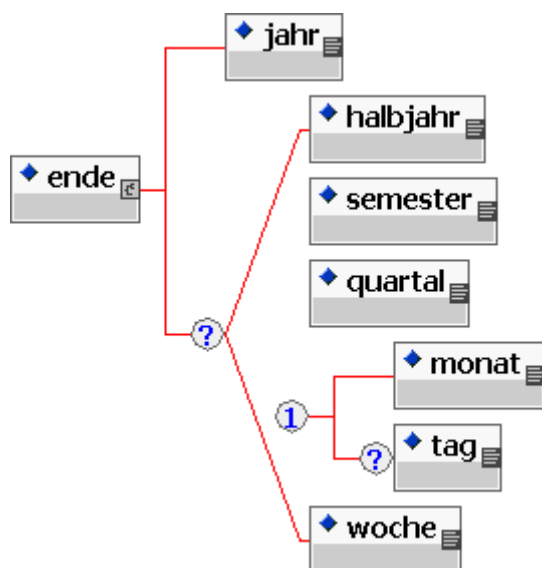
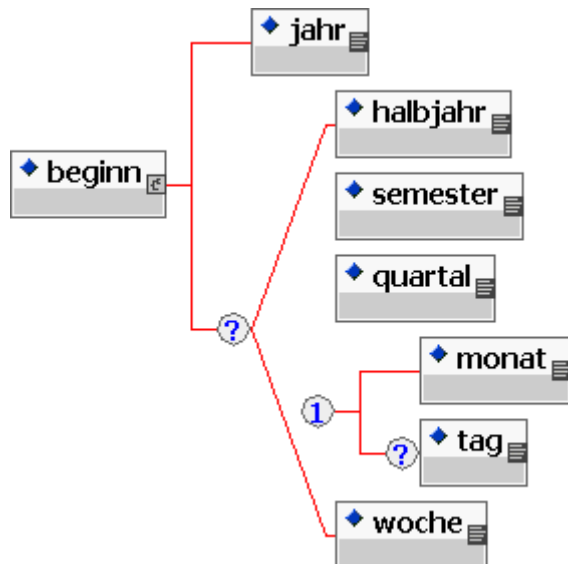
Details

Eine Zeile eines zeilenbasierten Datenmodells (s.2.7.2).

3.4. Erhebung, Berichtszeitraum

3.4.1. Elemente <beginn>, <ende>

Element	<beginn>, <ende>	
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

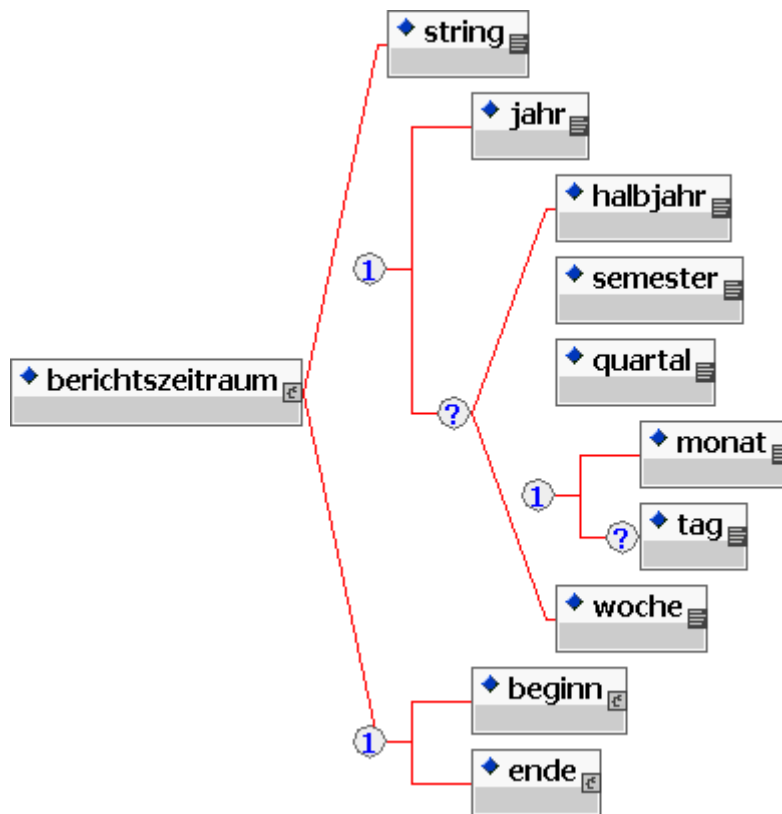


Details

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

3.4.2. Element <berichtszeitraum>

Element	<berichtszeitraum>		
Datentyp:	<i>element-content</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



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 November 2001:

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

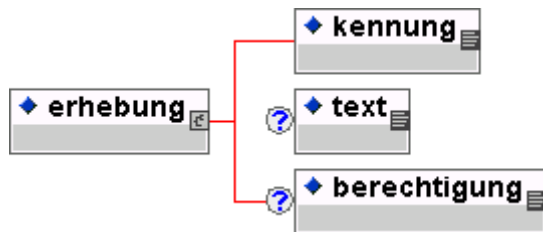
Der Berichtszeitraum kann auch eine formatierte Zeichenkette sein:

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

(s. 3.9.5, Element <string>). Die Festlegung zulässiger Datumsangaben und weitere Regeln (invertierte Spannen,..) sind Sache der Anwendung.

3.4.3. Element <erhebung>

Element	<erhebung>	
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

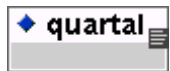


Details

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.

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

Element	<jahr>, <halbjahr>, <semester>, <quartal>, <monat>, <woche>, <tag>	
Status:	neu (nur <semester>)	≥ 2.0
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



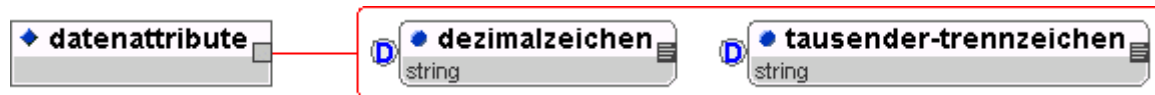
Details

Die Elemente sind Bestandteile von Datumsangaben. Der Wertebereich der Elemente entspricht denen in formatierten Datumsangaben (2.5.7.1).

3.5. Organisatorische und technische Informationen

3.5.1. Element <datenattribute>

Element	<datenattribute>	
Datentyp:	<i>empty-element</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



Attribute:

dezimalzeichen

S Datentyp, Default, Werte

D *char.*' , ' ; Das als Dezimalpunkt verwendete Zeichen.

tausender-trennzeichen

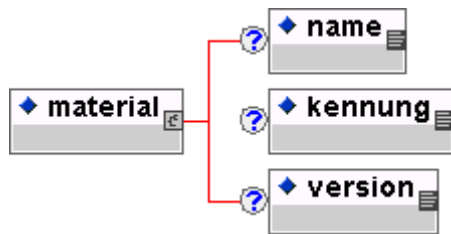
D *char.*' . ' ; Das als Tausender-Trennzeichen verwendete Zeichen.

Details

Das Element <datenattribute> beschreibt allgemeine Eigenschaften von Werten (<wert>) in einfachen (<mm>) und Ordnungsmerkmalen (<omm>). Für numerische Werte können Dezimalzeichen und Tausender-Trennzeichen abweichend vom Default bestimmt werden.

3.5.2. Element <material>

Element	<material>	
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



Details

Bezeichnung und Version des Liefermaterials.

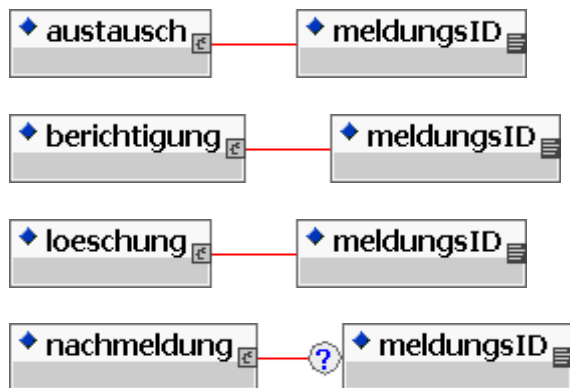
3.6. Datensegment

Ein Datensegment kapselt die Meldungsdatensätze (2.3.3).

3.6.1. Elemente <austausch>, <berichtigung>, <loeschung>, <nachmeldung>

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

Status:	neu	≥ 2.0
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

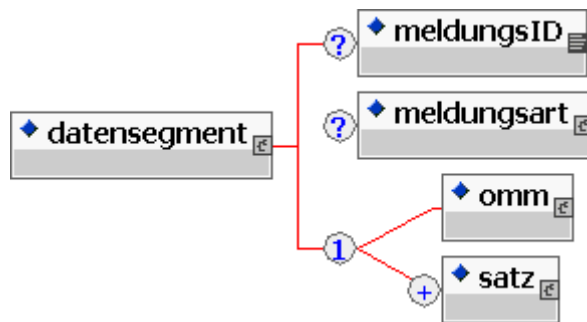


Details

Funktionselemente mit Bezug auf eine vorausgegangene Meldung (3.6.3).

3.6.2. Element <datensegment>

Element	<datensegment>	
Status:	neu	≥ 2.0
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



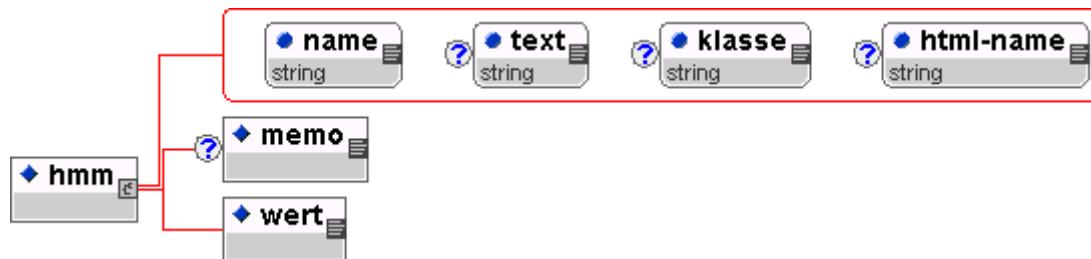
Details

Ein Datensegment kapselt die zu einer Meldung gehörenden Meldungsdatensätze und ist im Segmentbaum ein Blattknoten. Es schließt den Segmentpfad für eine Meldung ab und stellt die Verbindung zwischen Metadaten und Rohdaten her. An einem Datensegmentknoten ist der statistische Kontext für eine Meldung eindeutig; die Meldungs-ID und die Meldungsart sowie Datum und Uhrzeit der Meldungserzeugung können hier abgelegt werden.

Achtung: Meldungs-ID, Meldungsart, Datum und Uhrzeit sollten, wenn vorhanden, stets aus dem ursprünglichen Meldungsdocument übernommen werden; sind diese Werte einmal gesetzt, sollten sie in nachfolgenden Verarbeitungsstufen, die neue Dokumentinstanzen erzeugen, nicht überschrieben werden.

3.6.3. Element <hmm>

Element	<hmm>		
Datentyp:	<i>element-content</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



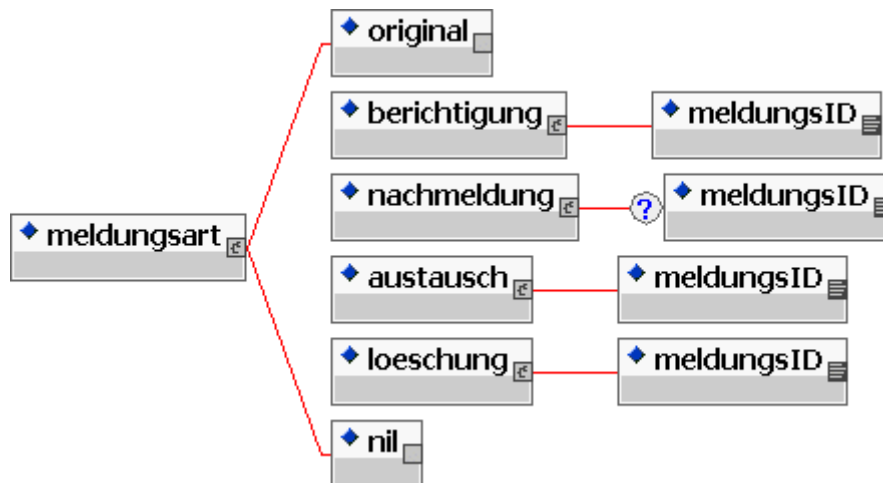
Attribute:	S	Datentyp, Default, Werte
name	M	<i>char.</i> ; Name des Hilfsmerkmals
text	?	<i>char.</i> ; Beschreibung des Hilfsmerkmals
klasse	?	<i>char.</i> ; Klasse oder Klassifikation
html-name	?	<i>char.</i> ; Name des (HTML-)Eingabefeldes

Details

Der Elementtyp <hmm> beschreibt ein statistisches Hilfsmerkmal (s. 2.5.2.3). Hilfsmerkmale sind meldungsglobal und haben einen für die ganze Meldung gültigen Wert. Das Attribut *name* identifiziert das Merkmal und muss immer angegeben werden.

3.6.4. Element <meldungsart>

Element	<meldungsart>	
Status:	neu	≥ 2.0
Datentyp:	element-content	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



Details

Mit der Meldungsart wird angegeben, für welchen Zweck die Meldung erfolgt. Außer Originallieferungen von Daten sind dies z.B. Korrekturen und Löschungen. Der Zweck einer Meldung wird durch das Element <meldungsart> ausgedrückt, welches ein entsprechendes Funktionselement enthält. Ist ein Bezug zu einer vorausgegangenen Meldung notwendig oder möglich, erfolgt dieser in Form einer Meldungs-ID, die dem Funktionselement nachgeordnet ist. Das Funktionselement <nil> unterdrückt die Verarbeitung einer Meldung.

3.6.5. Element <meldungsID>

Element	<meldungsID>	
Status:	neu	≥ 2.0
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



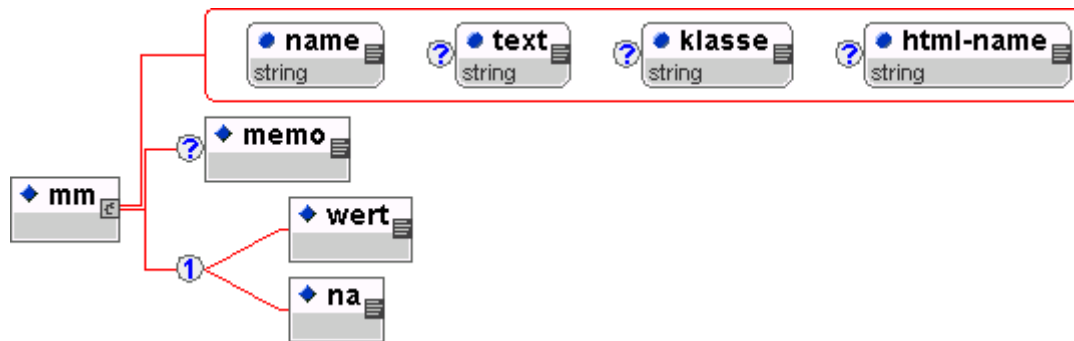
Attribute: **S** **Datentyp, Default, Werte**
 klasse ? *char*; Klasse oder Klassifikation

Details

Mit dem Element <meldungsID> kein 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.

3.6.6. Element <mm>

Element	<mm>		
Datentyp:	<i>element-content</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



Attribute:	S	Datentyp, <u>Default</u> , Werte
name	M	<i>char.</i> ; Name des Merkmals
text	?	<i>char.</i> ; Beschreibung des Merkmals
klasse	?	<i>char.</i> ; Klasse oder Klassifikation
html-name	?	<i>char.</i> ; Name des (HTML-)Eingabefeldes

Details

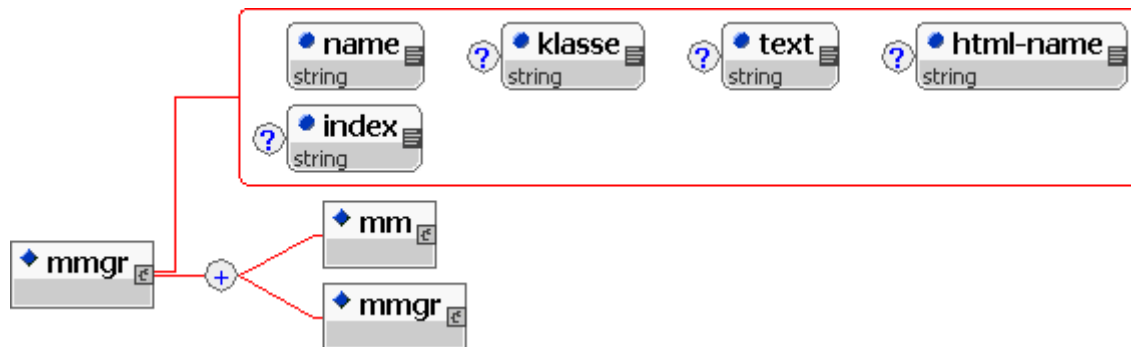
<mm> beschreibt ein statistisches Merkmal, das mit einem Wert (<wert>) oder einer expliziten Nichtantwort (<na>) verbunden ist. Das Attribut `name` identifiziert das Merkmal und muss immer angegeben werden; `text` erlaubt eine textuelle Merkmalsbeschreibung.

Beispiel:

```
<mm name="bruttogehaelt">
  <wert einheit="euro">2530</wert>
</mm>
```

3.6.7. Element <mmgr>

Element	<mmgr>		
Datentyp:	<i>element-content</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



Attribute:	S	Datentyp, <u>Default</u> , Werte
name	M	<i>char.</i> ; Name der Merkmalsgruppe
text	?	<i>char.</i> ; Beschreibung der Merkmalsgruppe
klasse	?	<i>char.</i> ; Klasse oder Klassifikation
html-name	?	<i>char.</i> ; Name des (HTML-)Eingabefeldes
index	?	<i>index.</i> ; Modus der Indexbildung

Details

Einfache Merkmale (<mm>) und Merkmalsgruppen selbst können zu einer wiederhol- und indizierbaren Merkmalsgruppe (<mmgr>) zusammengefasst werden. Das Attribut *name* identifiziert die Merkmalsgruppe und muss immer angegeben werden; *text* erlaubt eine zusätzliche textuelle Merkmalsbeschreibung. *index* definiert den Modus der Indexgewinnung (s. 2.5.3).

Ab Version 2.0 können Merkmalsgruppen geschachtelt werden (s. 2.5.3.2).

3.6.8. Element <na>

Element	<na>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



Details

<na> steht für eine explizite Nichtantwort. Der Elementinhalt darf beliebigen Text enthalten.

3.6.9. Elemente <nil>, <original>

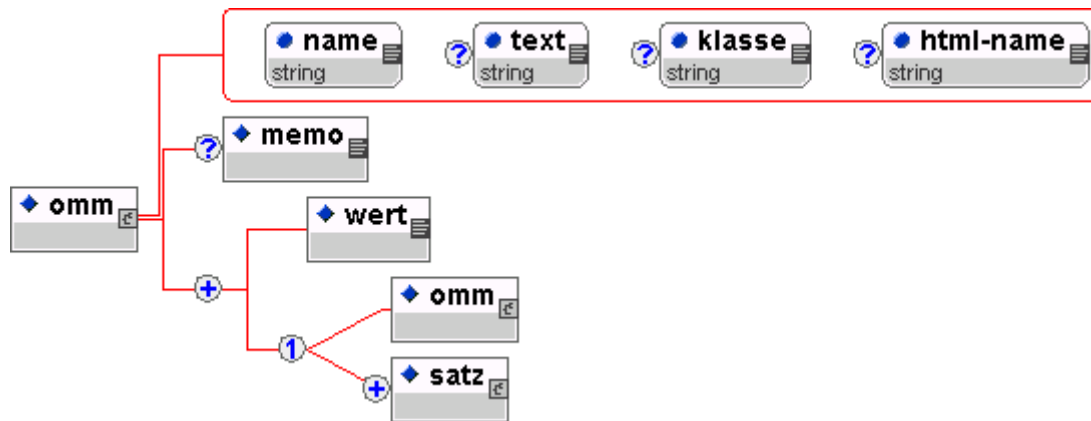
Element	<nil>, <original>	
Status:	<i>neu</i>	≥ 2.0
Datentyp:	<i>empty-element</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

**Details**

Funktionselemente *ohne* Bezug auf eine vorausgegangene Meldung (3.6.3).

3.6.10. Element <omm>

Element	<omm>		
Datentyp:	<i>element-content</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



Attribute:	S	Datentyp, Default, Werte
name	M	<i>char.</i> ; Name des Merkmals
text	?	<i>char.</i> ; Beschreibung des Merkmals
klasse	?	<i>char.</i> ; Klasse oder Klassifikation
html-name	?	<i>char.</i> ; Name des (HTML-)Eingabefeldes

Details

<omm> erlaubt es, Sätze an hierarchisch angeordnete Ordnungsmerkmale „anzuhängen“. Einem Ordnungsmerkmal dürfen beliebig viele Werte zugeordnet werden (<wert>). Zwischen zwei Werten steht ein weiteres, nachgeordnetes Ordnungsmerkmal *oder* eine Folge von Datensätzen (<satz>). Es ist ein Fehler, wenn es innerhalb eines Datensegmentes (<datensegment>) unterschiedlich lange Pfade über die Ordnungsmerkmale zu den Datensätzen gibt (unterschiedlich tief geschachtelte Ordnungsmerkmale) oder auf einer Hierarchiestufe Ordnungsmerkmale mit unterschiedlichem Namen vorhanden sind. Diese Einschränkungen sind nicht mit den Mitteln einer XML-DTD beschreibbar und müssen daher von der Anwendung kontrolliert werden

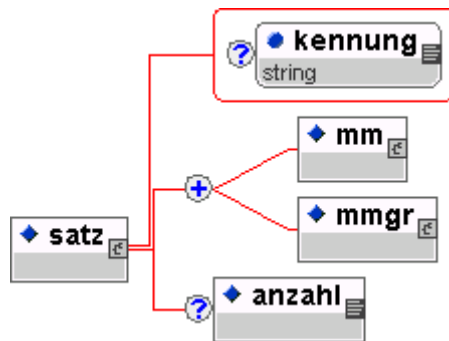
Das Attribut `name` identifiziert das Merkmal und muss immer angegeben werden; `text` erlaubt eine zusätzliche textuelle Merkmalsbeschreibung.

Beispiel:

```
<omm name="kh-nr" klasse="khv-2003">
  <wert>1200</wert>
  <satz>...</satz>
  <satz>...</satz>
  <wert>1300</wert>
  <satz>...</satz>
  <satz>...</satz>
  <!-- usw. -->
</omm>
```

3.6.11. Element <satz>

Element	<satz>	
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



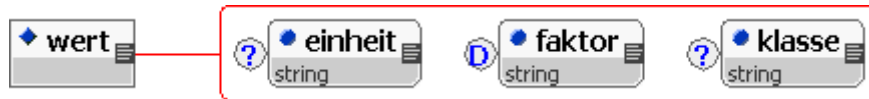
Attribute:	S Datentyp, <u>Default</u> , Werte
kennung	? <i>char.</i> ; Eine optionale Satzkennung

Details

Ein Satz besteht aus mindestens einem Wertmerkmal (<mm>) und einer optionalen Anzahlangabe (<anzahl>) für Kontrollzwecke. Das Attribut *kennung* ist eine Möglichkeit, Sätze unterschiedlichen Typs (Inhalts und Aufbaus) zu kennzeichnen. Anwendungen sind frei, auch andere Mechanismen der Satztyperkennung zu implementieren.

3.6.12. Element <wert>

Element	<wert>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



Attribute:

einheit
faktor
klasse

S Datentyp, Default, Werte

D *char*, Maßeinheit

D *int*₁; Faktor. Mit dem die Maßeinheit multipliziert wird.

? *char*, Klasse, wenn es für die Maßeinheit eine Klassifikation gibt, z.B. eine ISO- oder DIN-Norm.

Details

Die Maßeinheit eines Wertes kann bei Bedarf mit dem Attribut `einheit` angegeben werden. Das Attribut `faktor` bietet die Möglichkeit, einen Wert als n-faches bzw. n-ten Bruchteil der Maßeinheit anzugeben. Gibt es für die Maßeinheit eine Klassifikation, beispielsweise eine ISO-Norm, kann diese mit dem Attribut `x` angegeben werden.

Grundsätzlich ist die Angabe von Maßeinheiten und Faktoren nur in Ausnahmefällen sinnvoll. Werte sollten bereits in der für die Erhebung festgelegten Maßeinheit geliefert werden. Davon sollte nur abgewichen werden, wenn dies ausdrücklich erlaubt ist, da dann eine Umrechnung erfolgen muss; dies wird jedoch in den allermeisten Fällen nicht vorgesehen sein.

3.7. Verarbeitungssteuerung

3.7.1. Element <empfangsbestaetigung>

Element	<empfangsbestaetigung>		
Datentyp:	<i>empty-element</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-

The diagram illustrates the structure of the <empfangsbestaetigung> element. It is shown as a box with a blue diamond icon and the text 'empfangsbestaetigung'. A red line connects this box to a larger box containing three attribute boxes: 'versandart' (enumeration), 'format' (enumeration), and 'betreff' (string). Each attribute box has a blue circle icon and a list icon. The 'betreff' box also has a question mark icon.

Attribute: S Datentyp, Default, Werte

versandart	D	<i>enumeration:default</i> email post; Die Versandart
format	D	<i>enumeration:default</i> xml text; Das Antwortformat
betreff	?	<i>char.</i> ; Eine beliebige Betreffangabe

Details

Das Vorhandensein des Elementes <empfangsbestaetigung> zeigt an, dass der Absender eine Empfangsbestätigung verlangt. Versandart und Versandformat (nur bei E-Mail-Versand: XML-Nachricht als Anhang oder automatisch generierter Text) können bestimmt, eine Betreffangabe vorgegeben werden; s. 2.4.9.2.

3.7.2. Element <optionen>

Element	<optionen>	
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

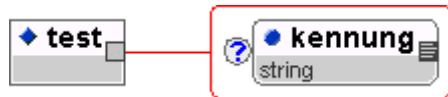


Details

Einstellungen und Benutzeroptionen, s 2.4.9.

3.7.3. Element <test>

Element	<test>		
Datentyp:	<i>empty-element</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



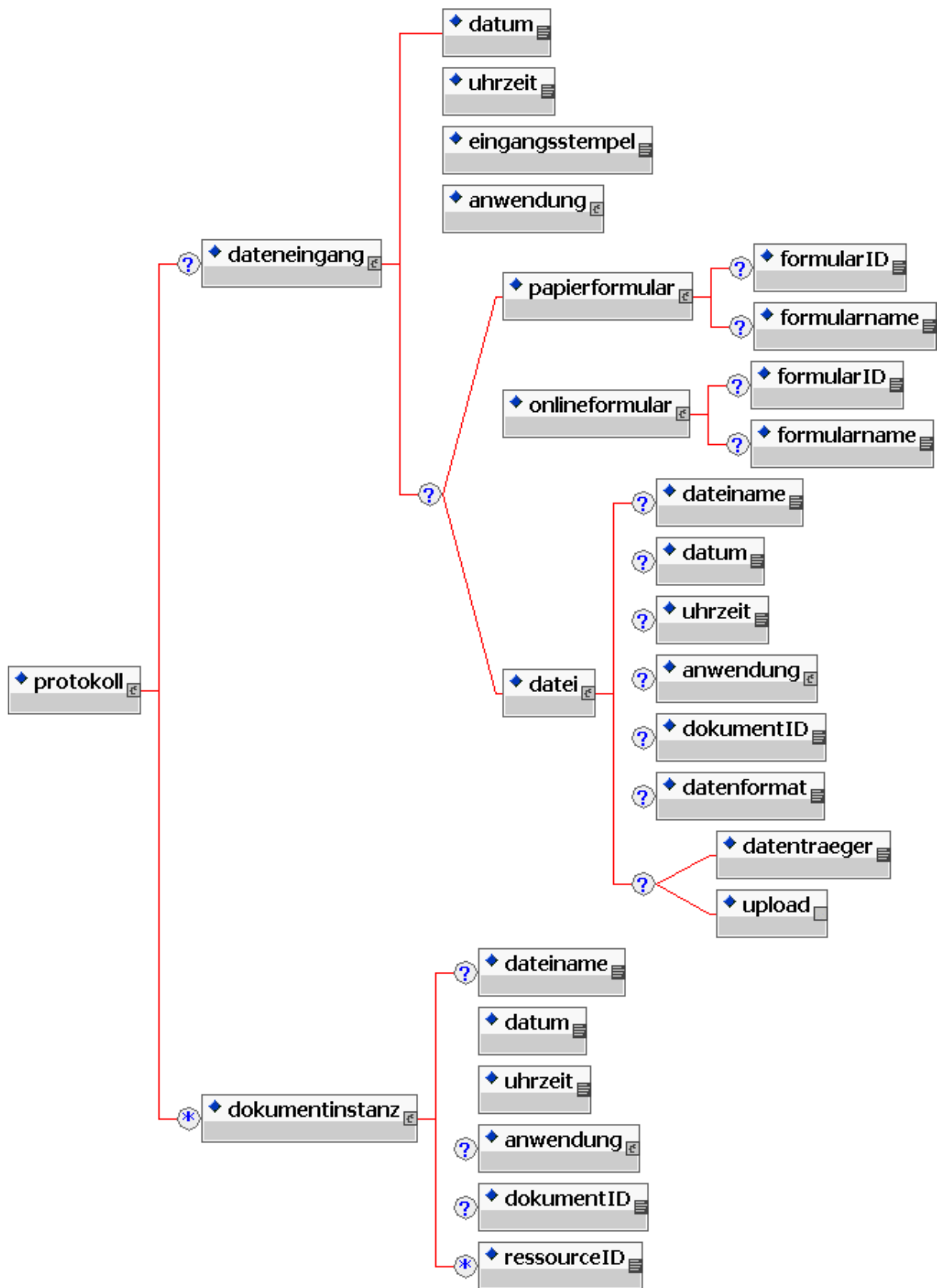
Attribute:	S	Datentyp, <u>Default</u>, Werte
kennung	?	<i>char.</i> ; Eine optionale Testkennung

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.

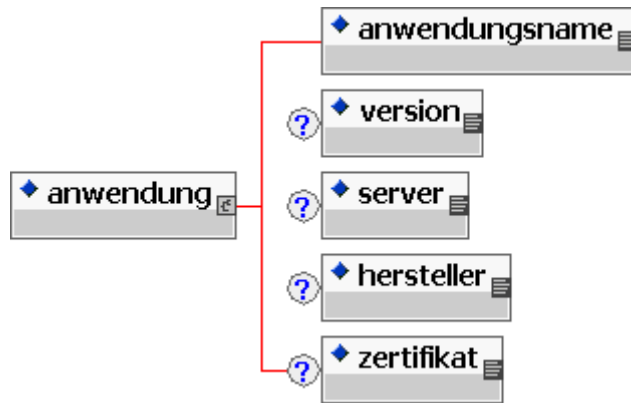
3.8. Protokollierung

3.8.1. Übersicht



3.8.2. Element <anwendung>

Element	<anwendung>	
Status:	neu	≥ 2.0
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

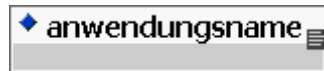


Details

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

3.8.3. Element <anwendungsname>

Element	<anwendungsname>	
Status:	neu	≥ 2.0
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

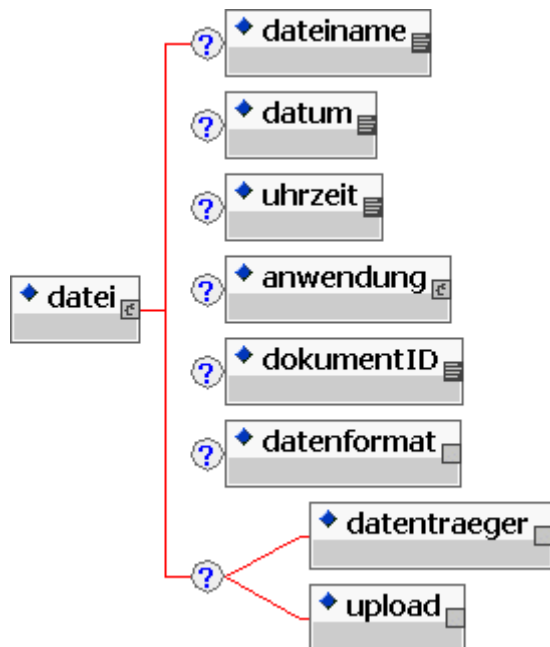


Details

Der Name einer (Software-)Anwendung.

3.8.4. Element <datei>

Element	<datei>	
Status:	neu	≥ 2.0
Datentyp:	<i>empty-element</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



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 erzeugt (<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/RAW-Instanz wird im Element <datenformat> durch Eintrag des Wertes `datml` im Attribut `datenformattyp` dokumentiert. Enthält die DatML/RAW-Instanz ein Element <dokumentinstanz> (s. 3.8.10), dann werden – sofern vorhanden – dessen Kindelemente <dateiname>, <datum>, <uhrzeit>, <dokumentID> und <anwendung> in das Element <datei> übernommen.

3.8.5. Element <dateiname>

Element	<dateiname>	
Status:	neu	≥ 2.0
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



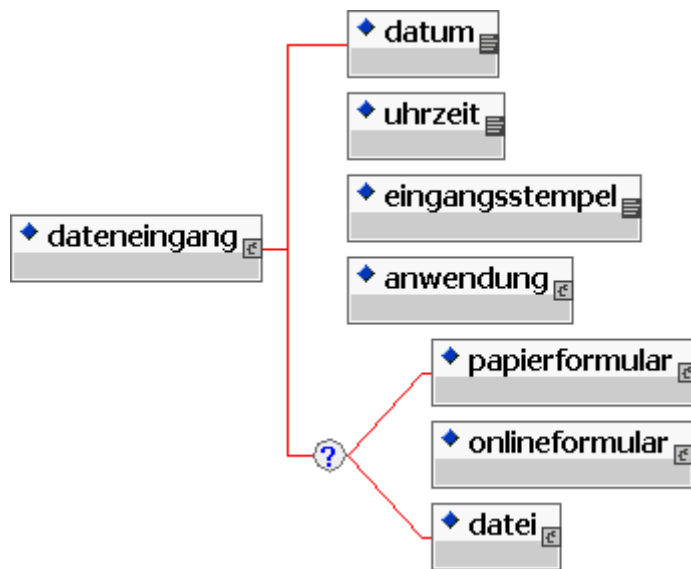
Details

Der Name einer Datei.

3.8.6. Element <dateneingang>

HINWEIS: <dateneingang> ersetzt den Elementtyp <eingang>, der ab Version 2.0 nur noch aus Kompatibilitätsgründen unterstützt wird.

Element	<dateneingang>	
Status:	neu	≥ 2.0
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



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 Dateneinganges, 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/RAW-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>
  <formulardname>FB-23111-2003</formulardname>
  <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>
```

3.8.7. Element <datenformat>

Element	<datenformat>	
Status:	neu	≥ 2.0
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



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

typ

D *enumeration:*

- undefiniert
- datml
- xml
- datensatz
- edifact

Typ des Datenformates; der Wert `datml` zeigt an, dass eine DatML/RAW-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>
```

3.8.8. Element <datentraeger>

Element	<datentraeger>	
Status:	neu	≥ 2.0
Datentyp:	<i>empty-element</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



Attribute: S Datentyp, Default, Werte

`typ` D *enumeration*: undefiniert | diskette | cdrom | dvd |
kassette | magnetband
Typ des Datenträgers

Details

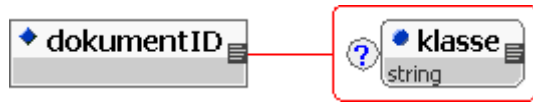
<datentraeger> gibt den Namen und Typ des verwendeten Datenträgers an.

Beispiel:

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

3.8.9. Element <dokumentID>

Element	<dokumentID>	
Status:	neu	≥ 2.0
Datentyp:	<i>empty-element</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



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

klasse D *char.default*; Optionale Klasse oder Klassifikation

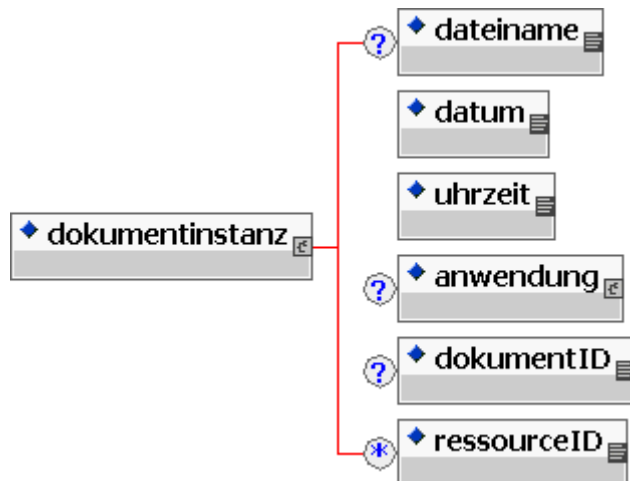
Details

<dokumentID> enthält eine Angabe für die (eindeutige) Identifizierung einer DatML/RAW-Dokumentinstanz.

3.8.10. Element <dokumentinstanz>

HINWEIS: <dokumentinstanz> ersetzt den Elementtyp <instanz>, der ab Version 2.0 nur noch aus Kompatibilitätsgründen unterstützt wird.

Element	<dokumentinstanz>	
Status:	<i>neu</i>	≥ 2.0
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



Details

<dokumentinstanz> erlaubt die Dokumentierung von DatML/RAW-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.

3.8.11. Element <eingangsstempel>

Element	<eingangsstempel>	
Status:	neu	≥ 2.0
Datentyp:	<i>empty-element</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



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

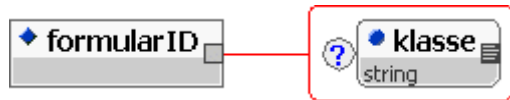
klasse D *char.default*; Optionale Klasse oder Klassifikation

Details

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

3.8.12. Element <formularID>

Element	<formularID>	
Status:	neu	≥ 2.0
Datentyp:	<i>empty-element</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

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

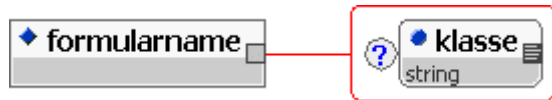
klasse D *char.default*; Optionale Klasse oder Klassifikation

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.

3.8.13. Element <formularname>

Element	<formularname>	
Status:	neu	≥ 2.0
Datentyp:	<i>empty-element</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

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

klasse D *char.default*; Optionale Klasse oder Klassifikation

Details

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

3.8.14. Element <hersteller>

Element	<hersteller>	
Status:	neu	≥ 2.0
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

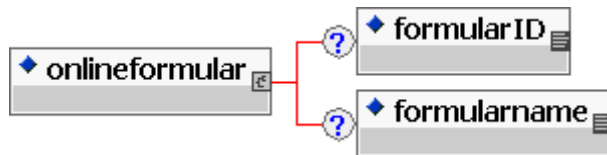


Details

Die Bezeichnung des Herstellers einer Anwendung.

3.8.15. Element `<onlineformular>`

Element	<code><onlineformular></code>	
Status:	neu	≥ 2.0
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

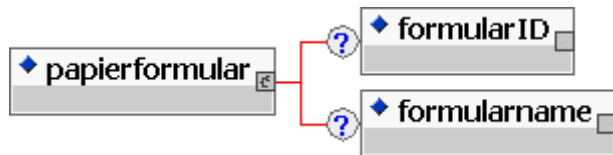


Details

`<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.

3.8.16. Element <papierformular>

Element	<papierformular>	
Status:	neu	≥ 2.0
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

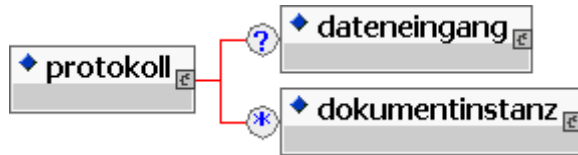


Details

<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.

3.8.17. Element <protokoll>

Element	<protokoll>		
Datentyp:	<i>element-content</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



Details

<protokoll> erlaubt die die Dokumentation des Dateneingangs (<dateneingang>) und beliebig vieler DatML/RAW-Dokumentinstanzen (<dokumentinstanz>).

3.8.18. Element <ressourceID>

Element	<ressourceID>	
Status:	neu	≥ 2.0
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

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

klasse D *char*.default; Optionale Klasse oder Klassifikation

Details

Die Kennzeichnung einer für die Dokumentgenerierung verwendeten Ressource.

3.8.19. Element <server>

Element	<server>	
Status:	neu	≥ 2.0
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



Details

Die Bezeichnung eines Servers.

3.8.20. Element <upload>

Element	<upload>	
Status:	neu	≥ 2.0
Datentyp:	<i>empty-element</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



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.

3.8.21. Element <zertifikat>

Element	<zertifikat	
Status:	neu	≥ 2.0
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



Attribute: **S** **Datentyp, Default, Werte**
klasse **D** *char.default*; Optionale Klasse oder Klassifikation

Details

Ein Zertifikatsschlüssel. Anwendungen können die Angabe eines Zertifikatsschlüssels vorschreiben.

3.9. Sonstige Elemente

3.9.1. Element <anzahl>

Element	<anzahl>	
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

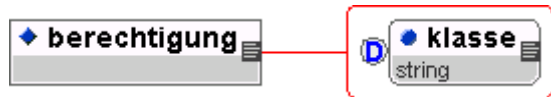


Details

Ein Kontrollwert für die Anzahlprüfung vorausgegangener Elemente.

3.9.2. Element <berechtigung>

Element	<berechtigung>		
Datentyp:	<i>char</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD:	-



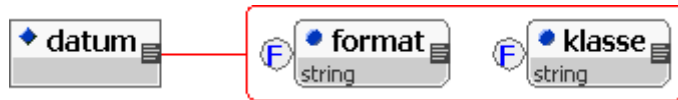
Attribute: **S** **Datentyp, Default, Werte**
klasse D *char*.default; Optionale Klasse oder Klassifikation

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.).

3.9.3. Element <datum>

Element	<datum>	
Status:	neu	≥ 2.0
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



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

Details

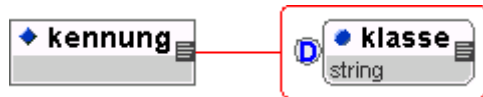
Ein Datum in Form eines Strings; Formatschema und Formatklasse sind fest vorgegeben (2.5.7.1).

3.9.4. Element <kennung>

Element <kennung>

Datentyp: *char*

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



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

klasse D *char*.default; Optionale Klasse oder Klassifikation

Details

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

3.9.5. Element `<memo>`

Element	<code><memo></code>	
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -

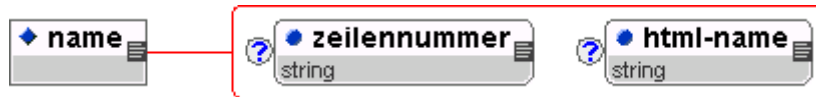


Details

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

3.9.6. Element <name>

Element	<name>	
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilennummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

Ein beliebiger Name, dessen Bedeutung kontextabhängig ist:

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

3.9.7. Element `<string>`

Element	<code><string></code>	
Datentyp:	<i>char</i>	
Namespace:	<code>http://www.destatis.de/schema/datml-raw/2.0/de</code>	DTD: -



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

Details

`<string>` ermöglicht die Angabe formatierter Zeichenketten. Derzeitig ist nur die Verwendung von Datumsangaben definiert (s. 2.5.7).

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

3.9.8. Element <text>

Element	<text>	
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



Details

Ein beliebiger Text mit kontextabhängiger Bedeutung.

3.9.9. Element <uhrzeit>

Element	<uhrzeit>	
Status:	neu	≥ 2.0
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



Attribute:	S	Datentyp, <u>Default</u> , Werte
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 (2.5.7.1). Wird die Uhrzeit nicht sekunden- oder sogar minutengenau angegeben, sind die entsprechenden Felder mit Nullen zu belegen.

3.9.10. Element `<version>`

Element	<code><version></code>
Datentyp:	<i>char</i>
Namespace:	<code>http://www.destatis.de/schema/datml-raw/2.0/de</code> DTD: -



Details

Eine Versionsangabe mit kontextabhängiger Bedeutung.

3.9.11. Element <zusatz>

Element	<zusatz>	
Datentyp:	<i>char</i>	
Namespace:	http://www.destatis.de/schema/datml-raw/2.0/de	DTD: -



Attribute:	S	Datentyp, <u>Default</u> , Werte
zeilennummer	?	<i>int</i> , Die Zeilennummer der Information in einem zeilenbasierten Datenmodell,
html-name	?	<i>char</i> , Der Name des (HTML-)Formularfeldes, in welches die Information eingegeben wurde.

Details

Ein Zusatz mit kontextabhängiger Bedeutung als generische Erweiterung eines Datenmodells.

4. Änderungen gegenüber Version 1.0.2

HINWEIS: Abwärtsinkompatible Änderungen können dazu führen, dass auf älteren Versionen basierende Dokumente nicht gegen eine neuere Version des Dokumenttyps validierbar sind. Dies betrifft nur Dokumente, die tatsächlich nicht mehr unterstützte Eigenschaften älterer Versionen verwenden. Da DatML-RAW-Dokumente die verwendete Version des Dokumenttyps spezifizieren bzw. explizit referenzieren, sollten sie auch nur gegen die angegebene bzw. referenzierte Version validiert werden.

4.1. Kompatible Änderungen

4.1.1. Dokument, Nachricht, Segment

#	Änderung	
A1-2.0	Art	Neuer Elementtyp <nachrichtenID>
	Elemente	<nachricht>
	Details	Als optionales Kindelement in <nachricht>

4.1.2. Teilnehmer, Identifikation, Kontaktangaben

#	Änderung	
A2-2.0	Art	Neuer Elementtyp <externeIdentifikation>
	Elemente	<absender>, <berichtsempfaenger>, <berichtspflichtiger>, <empfaenger>
	Details	Als optionales Kindelement in den genannten Elementtypen zulässig.
A3-2.0	Art	Erweitertes Inhaltsmodell
	Elemente	<identifikation>
	Details	Neue Kindelemente <email>, <fax>, <telefon>, <url>
A4-2.0	Art	Erweitertes Inhaltsmodell
	Elemente	<kontakt>
	Details	Neues Kindelemente <adresse>
A5-2.0	Art	Neuer Elementtyp <hausnummer>
	Elemente	<adresse>
	Details	Als optionales Kindelement in <adresse>

4.1.3. Erhebung, Berichtszeitraum

#	Änderung	
A6-2.0	Art	Neuer Elementtyp <semester>
	Elemente	<berichtszeitraum>
	Details	Als zusätzliche Alternative zu <halbjahr>, <quartal>, <woche>

4.1.4. Datensegment

#	Änderung	
A7-2.0	Art	Erweitertes Inhaltsmodell
	Elemente	<mmgr>
	Details	Merkmalsgruppen dürfen geschachtelt sein
A8-2.0	Art	Neue Elementtypen <meldungsID> und <meldungsart>
	Elemente	<datensegment>
	Details	Als Kindelemente in den genannten Elementtypen, teilweise erforderlich.

4.1.5. Sonstige Elemente

#	Änderung	
A9-2.0	Art	Neue Elementtypen <datum> und <uhrzeit>
	Elemente	<datensegment>, <datei>, <dateneingang>, <dokumentinstanz>
	Details	Als Kindelemente in den genannten Elementtypen, teilweise erforderlich.
A10-2.0	Art	Neue Attribute zeilennummer und html-name
	Elemente	<zusatz>
	Details	-

4.2. Inkompatible Änderungen

4.2.1. Dokument, Nachricht, Segment

#	Änderung	
I1-2.0	Art	Feste Reihenfolge der Kindelemente
	Elemente	<nachricht>, <segment>
	Details	Die Reihenfolge der Kindelemente in diesem Elementtypen ist nicht mehr beliebig.

4.2.2. Protokollierung

#	Änderung	
I2-2.0	Art	Elementtyp ersetzt
	Elemente	<eingang>
	Details	Durch <dateneingang> ersetzt.
I3-2.0	Art	Elementtyp ersetzt
	Elemente	<instanz>
	Details	Durch <dokumentinstanz> ersetzt.

4.2.3. Daten

#	Änderung	
I4-2.0	Art	Attribute ersetzt
	Elemente	<wert>
	Details	Die Attribute wurden wie folgt geändert: einheit ersetzt einheit-basis (sowie einheit-text), faktor ersetzt einheit-faktor, einheit-text entfällt, klasse ist neu.

4.3. Sonstige Änderungen

4.3.1. Allgemein

#	Änderung	
S1-2.0	Art	Erweiterte Formatklasse datum
	Elemente	<string>
	Details	Die Formatklasse datum wurde um Semesterangaben erweitert (2.5.7.1).

4.3.2. Protokollierung

#	Änderung	
S2-2.0	Art	Neue Elemente: <anwendung>, <anwendungsname>, <dateiname>, <datenformat>, <datentraeger>, <dokumentID>, <datei>, <eingangstempel>, <formularID>, <formulardname>, <hersteller>, <onlineformular>, <papierformular>, <ressourceID>, <server>, <upload>, <zertifikat>
	Elemente	<dateneingang>, <dokumentinstanz>
	Details	Die Elementtypen implementieren das Inhaltsmodell der neuen Elementtypen <dateneingang> und <dokumentinstanz>

5. Beispieldokument *[nicht-normativ]*

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

```
<?xml version="1.0" encoding="UTF-8"?>
<DatML-RAW-D version="2.0" xmlns="http://www.destatis.de/schema/datml-raw/2.0/de">
  <!-- Beispieldatei zum XML-Lieferformat DatML/RAW fuer
        die Gewerbeanzeigenstatistik

        Die Beispiele sind der EDIFACT-Dokumentation zur
        Gewerbeanzeigenstatistik entnommen, Abschnitt 5 Musterbögen.

        WICHTIG:
        Alle Kommentare und Einrueckungen sind optional und dienen
        hier nur zum besseren Verstaendnis bzw. Lesbarkeit. Für
        Originallieferungen kann auf Kommentierung verzichtet wer-
        den, Einrueckung ist sinnvoll, wenn Dokumente mit ueblichen
        Editoren bearbeit werden muessen.
  -->
  <!-- Optionen
        Der Absender kennzeichnet die Lieferung als Testlieferung.
  -->
  <optionen>
    <test kennung="100"/>
  </optionen>
  <!-- Protokoll
  -->
  <protokoll>
    <dateneingang>
      <datum>20041031</datum>
      <uhrzeit>152609</uhrzeit>
      <eingangsstempel/>
      <anwendung>
        <anwendungsname>IDEV 0.5 (Beta)</anwendungsname>
      </anwendung>
      <onlineformular>
        <formularname>FB-GAW-2005</formularname>
      </onlineformular>
    </dateneingang>
    <dokumentinstanz>
      <dateiname/>
      <datum>20041031</datum>
      <uhrzeit>154313</uhrzeit>
      <anwendung>
        <anwendungsname>IDEV 0.5 (Beta)</anwendungsname>
      </anwendung>
    </dokumentinstanz>
    <dokumentinstanz>
      <datum>20041108</datum>
      <uhrzeit>150904</uhrzeit>
      <anwendung>
        <anwendungsname>rawlraw2.xsl</anwendungsname>
        <version>1.0</version>
      </anwendung>
    </dokumentinstanz>
  </protokoll>
  <!-- Absenderangaben
        Kennung des Absenders
  -->
  <absender>
    <kennung klasse="MELDID">HB492949</kennung>
  </absender>
  <!-- Empfaengerangaben
        Name und Adresse des Empfängers
  -->
  <empfaenger>
    <kennung klasse="STAID">04</kennung>
  </empfaenger>
  <!-- Nachricht mit den Datensätzen
        Jeder Datensatz entspricht einem Berichtsfall der
        Gewerbeanzeigenstatistik
  -->
  <nachricht>
    <erhebung>
```

```
<kennung klasse="ERHID">52311.1</kennung>
</erhebung>
<berichtszeitraum>
  <jahr>2005</jahr>
  <monat>01</monat>
</berichtszeitraum>
<hmm name="GemeindeName">
  <wert>Bremen</wert>
</hmm>
<hmm name="GemeindeSchluessel">
  <wert>04011000</wert>
</hmm>
<hmm name="AGSSchluesselVerzeichnis">
  <wert>AGS092002</wert>
</hmm>
<datensegment>
  <satz>
    <mm name="MeldungArt">
      <wert>1</wert>
    </mm>
    <mm name="IstKorrektur">
      <wert>0</wert>
    </mm>
    <mm name="GemeindeSchluessel">
      <wert>04011000</wert>
    </mm>
    <mm name="AGSSchluesselVerzeichnis">
      <wert>AGS092002</wert>
    </mm>
    <mm name="GemeindeMeldungNr">
      <wert>0673456</wert>
    </mm>
    <mmgr name="Gewerbe">
      <mmgr name="Betrieb">
        <mm name="EingetragenerName">
          <wert>Beta Markt GmbH & Co. KG</wert>
        </mm>
        <mm name="EingetragenerNameZusatz">
          <wert>Vertriebsgesellschaft Bremen</wert>
        </mm>
        <mm name="EintragungNr">
          <wert>22871</wert>
        </mm>
        <mm name="EintragungOrt">
          <wert>Bremen</wert>
        </mm>
        <mm name="EintragungPLZ">
          <wert>28195</wert>
        </mm>
        <mm name="EintragungArt">
          <wert>A</wert>
        </mm>
        <mm name="GerichtSchluessel">
          <wert>041111</wert>
        </mm>
        <mm name="RGGSchluesselVerzeichnis">
          <wert>RGG012003</wert>
        </mm>
        <mm name="RechtsformSchluessel">
          <wert>230</wert>
        </mm>
        <mm name="RFMSchluesselVerzeichnis">
          <wert>RFM012003</wert>
        </mm>
      </mmgr>
    <mm name="AnzahlBeteiligte">
      <wert>3</wert>
    </mm>
    <mm name="AnzahlVollzeitBeschaeftigte">
      <wert>4</wert>
    </mm>
    <mm name="AnzahlTeilzeitBeschaeftigte">
      <wert>2</wert>
    </mm>
    <mm name="BetriebArtIndustrie">
      <wert>0</wert>
    </mm>
    <mm name="BetriebArtHandwerk">
```

```
<wert>0</wert>
</mm>
<mm name="BetriebArtHandel">
  <wert>1</wert>
</mm>
<mm name="BetriebArtSonstiges">
  <wert>0</wert>
</mm>
<mm name="NiederlassungArt">
  <wert>1</wert>
</mm>
<mm name="IstAutomatenAufsteller">
  <wert>0</wert>
</mm>
<mm name="IstReisegewerbe">
  <wert>0</wert>
</mm>
<mm name="IstNebenerwerb">
  <wert>0</wert>
</mm>
<mmgr name="Anschrift">
  <mm name="Strasse">
    <wert>Hanseatenstr.</wert>
  </mm>
  <mm name="HausNrVon">
    <wert>19</wert>
  </mm>
  <mm name="PLZ">
    <wert>28305</wert>
  </mm>
  <mm name="Ort">
    <wert>Bremen</wert>
  </mm>
</mmgr>
<mmgr name="BeteiligteFirma">
  <mmgr name="Betrieb">
    <mm name="EingetragenerName">
      <wert>Beta Markt GmbH</wert>
    </mm>
    <mm name="EintragungNr">
      <wert>77163</wert>
    </mm>
    <mm name="EintragungOrt">
      <wert>Hannover</wert>
    </mm>
    <mm name="EintragungPLZ">
      <wert>30175</wert>
    </mm>
    <mm name="EintragungArt">
      <wert>B</wert>
    </mm>
    <mm name="GerichtSchluessel">
      <wert>034316</wert>
    </mm>
    <mm name="RGGSchluesselVerzeichnis">
      <wert>RGG012003</wert>
    </mm>
  </mmgr>
  <mmgr name="BeteiligtePerson">
    <mm name="PersonFunktion">
      <wert>3</wert>
    </mm>
    <mm name="AnredeSchluessel">
      <wert>03</wert>
    </mm>
    <mm name="ANRSchluesselVerzeichnis">
      <wert>ANR111995</wert>
    </mm>
    <mm name="Familienname">
      <wert>Wolters</wert>
    </mm>
    <mm name="Vorname">
      <wert>Klaus</wert>
    </mm>
    <mm name="Geschlecht">
      <wert>1</wert>
    </mm>
    <mm name="Geburtsdatum">
```

```
<wert>02011948</wert>
</mm>
<mm name="Geburtsort">
  <wert>Uedem, Krs. Kleve</wert>
</mm>
<mm name="Geburtsland">
  <wert>Deutschland</wert>
</mm>
<mm name="Staatsangehoerigkeit">
  <wert>deutsch</wert>
</mm>
<mm name="StaatsangehoerigkeitSchluessel">
  <wert>000</wert>
</mm>
<mm name="SGSSchluesselVerzeichnis">
  <wert>SGS012003</wert>
</mm>
<mmgr name="Anschrift">
  <mm name="Strasse">
    <wert>Hartestr.</wert>
  </mm>
  <mm name="HausNrVon">
    <wert>15</wert>
  </mm>
  <mm name="PLZ">
    <wert>30539</wert>
  </mm>
  <mm name="Ort">
    <wert>Hannover</wert>
  </mm>
</mmgr>
</mmgr>
<!-- beteiligtePerson -->
<mmgr name="BeteiligtePerson">
  <mm name="PersonFunktion">
    <wert>3</wert>
  </mm>
  <mm name="AnredeSchluessel">
    <wert>02</wert>
  </mm>
  <mm name="ANRSchluesselVerzeichnis">
    <wert>ANR111995</wert>
  </mm>
  <mm name="Familiename">
    <wert>Schulz</wert>
  </mm>
  <mm name="Vorname">
    <wert>Jeanette</wert>
  </mm>
  <mm name="Geschlecht">
    <wert>2</wert>
  </mm>
  <mm name="Geburtsname">
    <wert>Molière</wert>
  </mm>
  <mm name="Geburtsdatum">
    <wert>30041952</wert>
  </mm>
  <mm name="Geburtsort">
    <wert>Paris</wert>
  </mm>
  <mm name="Geburtsland">
    <wert>Frankreich</wert>
  </mm>
  <mm name="Staatsangehoerigkeit">
    <wert>französisch</wert>
  </mm>
  <mm name="StaatsangehoerigkeitSchluessel">
    <wert>129</wert>
  </mm>
  <mm name="SGSSchluesselVerzeichnis">
    <wert>SGS012003</wert>
  </mm>
<mmgr name="Anschrift">
  <mm name="Strasse">
    <wert>Färberstr.</wert>
  </mm>
  <mm name="HausNrVon">
```

```

        <wert>15</wert>
      </mm>
      <mm name="PLZ">
        <wert>30453</wert>
      </mm>
      <mm name="Ort">
        <wert>Hannover</wert>
      </mm>
    </mmgr>
  </mmgr>
  <!-- beteiligtePerson -->
</mmgr>
  <!-- beteiligteFirma -->
</mmgr>
  <!-- gewerbe -->
  <mm name="GueltigAb">
    <wert>01012003</wert>
  </mm>
  <mm name="DatumMeldung">
    <wert>15012003</wert>
  </mm>
  <mmgr name="Taetigkeiten">
    <mm name="Taetigkeit">
      <wert>Einzelhandel mit Lebensmitteln und Gebrauchsgütern aller Art, Spirituosen,
Tabak- und Drogeriewaren</wert>
    </mm>
    <mm name="TaetigkeitSchluessel">
      <wert>52112</wert>
    </mm>
    <mm name="WZGSchluesselVerzeichnis">
      <wert>WZG012003</wert>
    </mm>
    <mm name="IstSchwerpunkt">
      <wert>1</wert>
    </mm>
  </mmgr>
  <mm name="GrundAnmeldungSchluessel">
    <wert>03</wert>
  </mm>
</satz>
<satz>
  <mm name="MeldungArt">
    <wert>1</wert>
  </mm>
  <mm name="IstKorrektur">
    <wert>0</wert>
  </mm>
  <mm name="GemeindeSchluessel">
    <wert>04011000</wert>
  </mm>
  <mm name="AGSSchluesselVerzeichnis">
    <wert>AGS092002</wert>
  </mm>
  <mm name="GemeindeMeldungNr">
    <wert>0673456</wert>
  </mm>
  <mmgr name="Gewerbe">
    <mmgr name="Betrieb">
      <mm name="EingetragenerName">
        <wert>Beta Markt GmbH & Co. KG</wert>
      </mm>
      <mm name="EingetragenerNameZusatz">
        <wert>Vertriebsgesellschaft Bremen</wert>
      </mm>
      <mm name="EintragungNr">
        <wert>22871</wert>
      </mm>
      <mm name="EintragungOrt">
        <wert>Bremen</wert>
      </mm>
      <mm name="EintragungPLZ">
        <wert>28195</wert>
      </mm>
      <mm name="EintragungArt">
        <wert>A</wert>
      </mm>
      <mm name="GerichtSchluessel">
        <wert>041111</wert>
      </mm>
    </mmgr>
  </mmgr>

```

```
</mm>
<mm name="RGGSchluesselVerzeichnis">
  <wert>RGG012003</wert>
</mm>
<mm name="RechtsformSchluessel">
  <wert>230</wert>
</mm>
<mm name="RFMSchluesselVerzeichnis">
  <wert>RFM012003</wert>
</mm>
</mmgr>
<mm name="AnzahlBeteiligte">
  <wert>3</wert>
</mm>
<mm name="AnzahlVollzeitBeschaeftigte">
  <wert>4</wert>
</mm>
<mm name="AnzahlTeilzeitBeschaeftigte">
  <wert>2</wert>
</mm>
<mm name="BetriebArtIndustrie">
  <wert>0</wert>
</mm>
<mm name="BetriebArtHandwerk">
  <wert>0</wert>
</mm>
<mm name="BetriebArtHandel">
  <wert>1</wert>
</mm>
<mm name="BetriebArtSonstiges">
  <wert>0</wert>
</mm>
<mm name="NiederlassungArt">
  <wert>1</wert>
</mm>
<mm name="IstAutomatenAufsteller">
  <wert>0</wert>
</mm>
<mm name="IstReisegewerbe">
  <wert>0</wert>
</mm>
<mm name="IstNebenerwerb">
  <wert>0</wert>
</mm>
<mmgr name="Anschrift">
  <mm name="Strasse">
    <wert>Hanseatenstr.</wert>
  </mm>
  <mm name="HausNrVon">
    <wert>19</wert>
  </mm>
  <mm name="PLZ">
    <wert>28305</wert>
  </mm>
  <mm name="Ort">
    <wert>Bremen</wert>
  </mm>
</mmgr>
<mmgr name="BeteiligteFirma">
  <mmgr name="Betrieb">
    <mm name="EingetragenerName">
      <wert>ABC-GmbH</wert>
    </mm>
    <mm name="EintragungNr">
      <wert>7815</wert>
    </mm>
    <mm name="EintragungOrt">
      <wert>Hannover</wert>
    </mm>
    <mm name="EintragungPLZ">
      <wert>30175</wert>
    </mm>
    <mm name="EintragungArt">
      <wert>B</wert>
    </mm>
    <mm name="GerichtSchluessel">
      <wert>034316</wert>
    </mm>
  </mmgr>
</mmgr>
```

```

    <mm name="RGGSchluesselVerzeichnis">
      <wert>RGG012003</wert>
    </mm>
  </mmgr>
<mmgr name="BeteiligtePerson">
  <mm name="PersonFunktion">
    <wert>3</wert>
  </mm>
  <mm name="AnredeSchluessel">
    <wert>03</wert>
  </mm>
  <mm name="ANRSchluesselVerzeichnis">
    <wert>ANR111995</wert>
  </mm>
  <mm name="Familiename">
    <wert>Steiger</wert>
  </mm>
  <mm name="Vorname">
    <wert>Martin</wert>
  </mm>
  <mm name="Geschlecht">
    <wert>1</wert>
  </mm>
  <mm name="Geburtsdatum">
    <wert>15121944</wert>
  </mm>
  <mm name="Geburtsort">
    <wert>Kirchdorf</wert>
  </mm>
  <mm name="Geburtsland">
    <wert>Deutschland</wert>
  </mm>
  <mm name="Staatsangehoerigkeit">
    <wert>deutsch</wert>
  </mm>
  <mm name="StaatsangehoerigkeitSchluessel">
    <wert>000</wert>
  </mm>
  <mm name="SGSSchluesselVerzeichnis">
    <wert>SGS012003</wert>
  </mm>
<mmgr name="Anschrift">
  <mm name="Strasse">
    <wert>Hansestr.</wert>
  </mm>
  <mm name="HausNrVon">
    <wert>14</wert>
  </mm>
  <mm name="PLZ">
    <wert>30419</wert>
  </mm>
  <mm name="Ort">
    <wert>Hannover</wert>
  </mm>
</mmgr>
</mmgr>
<!-- beteiligtePerson -->
<mmgr name="BeteiligtePerson">
  <mm name="PersonFunktion">
    <wert>3</wert>
  </mm>
  <mm name="AnredeSchluessel">
    <wert>03</wert>
  </mm>
  <mm name="ANRSchluesselVerzeichnis">
    <wert>ANR111995</wert>
  </mm>
  <mm name="Familiename">
    <wert>Müller</wert>
  </mm>
  <mm name="Vorname">
    <wert>Erwin</wert>
  </mm>
  <mm name="Geschlecht">
    <wert>1</wert>
  </mm>
  <mm name="Geburtsdatum">
    <wert>01051950</wert>

```

```

    </mm>
    <mm name="Geburtsort">
      <wert>Halle</wert>
    </mm>
    <mm name="Geburtsland">
      <wert>Deutschland</wert>
    </mm>
    <mm name="Staatsangehoerigkeit">
      <wert>deutsch</wert>
    </mm>
    <mm name="StaatsangehoerigkeitSchluessel">
      <wert>000</wert>
    </mm>
    <mm name="SGSSSchluesselVerzeichnis">
      <wert>SGS012003</wert>
    </mm>
    <mmgr name="Anschrift">
      <mm name="Strasse">
        <wert>Großweg</wert>
      </mm>
      <mm name="HausNrVon">
        <wert>3</wert>
      </mm>
      <mm name="PLZ">
        <wert>30457</wert>
      </mm>
      <mm name="Ort">
        <wert>Hannover</wert>
      </mm>
    </mmgr>
    <!-- beteiligtePerson -->
  </mmgr>
  <!-- beteiligteFirma -->
</mmgr>
<!-- gewerbe -->
<mm name="GueltigAb">
  <wert>01012003</wert>
</mm>
<mm name="DatumMeldung">
  <wert>17012003</wert>
</mm>
<mmgr name="Taetigkeiten">
  <mm name="Taetigkeit">
    <wert>Einzelhandel mit Lebensmitteln und Gebrauchsgütern aller Art, Spirituosen,
Tabak- und Drogeriewaren</wert>
  </mm>
  <mm name="TaetigkeitSchluessel">
    <wert>52112</wert>
  </mm>
  <mm name="WZGSchluesselVerzeichnis">
    <wert>WZG012003</wert>
  </mm>
  <mm name="IstSchwerpunkt">
    <wert>1</wert>
  </mm>
</mmgr>
<mm name="GrundAnmeldungSchluessel">
  <wert>03</wert>
</mm>
</satz>
<satz>
  <mm name="MeldungArt">
    <wert>1</wert>
  </mm>
  <mm name="IstKorrektur">
    <wert>0</wert>
  </mm>
  <mm name="GemeindeSchluessel">
    <wert>04011000</wert>
  </mm>
  <mm name="AGSSchluesselVerzeichnis">
    <wert>AGS092002</wert>
  </mm>
  <mm name="GemeindeMeldungNr">
    <wert>0673456</wert>
  </mm>
<mmgr name="Gewerbe">

```

```
<mmgr name="Betrieb">
  <mm name="EingetragenerName">
    <wert>Beta Markt GmbH & Co. KG</wert>
  </mm>
  <mm name="EingetragenerNameZusatz">
    <wert>Vertriebsgesellschaft Bremen</wert>
  </mm>
  <mm name="EintragungNr">
    <wert>22871</wert>
  </mm>
  <mm name="EintragungOrt">
    <wert>Bremen</wert>
  </mm>
  <mm name="EintragungPLZ">
    <wert>28195</wert>
  </mm>
  <mm name="EintragungArt">
    <wert>A</wert>
  </mm>
  <mm name="GerichtSchluessel">
    <wert>041111</wert>
  </mm>
  <mm name="RGGSchluesselVerzeichnis">
    <wert>RGG012003</wert>
  </mm>
  <mm name="RechtsformSchluessel">
    <wert>230</wert>
  </mm>
  <mm name="RFMSchluesselVerzeichnis">
    <wert>RFM012003</wert>
  </mm>
</mmgr>
<mm name="AnzahlBeteiligte">
  <wert>3</wert>
</mm>
<mm name="AnzahlVollzeitBeschaeftigte">
  <wert>4</wert>
</mm>
<mm name="AnzahlTeilzeitBeschaeftigte">
  <wert>2</wert>
</mm>
<mm name="BetriebArtIndustrie">
  <wert>0</wert>
</mm>
<mm name="BetriebArtHandwerk">
  <wert>0</wert>
</mm>
<mm name="BetriebArtHandel">
  <wert>1</wert>
</mm>
<mm name="BetriebArtSonstiges">
  <wert>0</wert>
</mm>
<mm name="NiederlassungArt">
  <wert>1</wert>
</mm>
<mm name="IstAutomatenAufsteller">
  <wert>0</wert>
</mm>
<mm name="IstReisegewerbe">
  <wert>0</wert>
</mm>
<mm name="IstNebenerwerb">
  <wert>0</wert>
</mm>
<mmgr name="Anschrift">
  <mm name="Strasse">
    <wert>Hanseatenstr.</wert>
  </mm>
  <mm name="HausNrVon">
    <wert>19</wert>
  </mm>
  <mm name="PLZ">
    <wert>28305</wert>
  </mm>
  <mm name="Ort">
    <wert>Bremen</wert>
  </mm>
</mmgr>
```

```

</mmgr>
<mmgr name="BeteiligtePerson">
  <mm name="PersonFunktion">
    <wert>2</wert>
  </mm>
  <mm name="AnredeSchluessel">
    <wert>03</wert>
  </mm>
  <mm name="ANRSchluesselVerzeichnis">
    <wert>ANR111995</wert>
  </mm>
  <mm name="Familiename">
    <wert>Schmitt</wert>
  </mm>
  <mm name="Vorname">
    <wert>Anton</wert>
  </mm>
  <mm name="Geschlecht">
    <wert>1</wert>
  </mm>
  <mm name="Geburtsdatum">
    <wert>01031940</wert>
  </mm>
  <mm name="Geburtsort">
    <wert>Hamburg</wert>
  </mm>
  <mm name="Geburtsland">
    <wert>Deutschland</wert>
  </mm>
  <mm name="Staatsangehoerigkeit">
    <wert>deutsch</wert>
  </mm>
  <mm name="StaatsangehoerigkeitSchluessel">
    <wert>000</wert>
  </mm>
  <mm name="SGSSchluesselVerzeichnis">
    <wert>SGS012003</wert>
  </mm>
  <mmgr name="Anschrift">
    <mm name="Strasse">
      <wert>Nordseestr.</wert>
    </mm>
    <mm name="HausNrVon">
      <wert>3</wert>
    </mm>
    <mm name="PLZ">
      <wert>28333</wert>
    </mm>
    <mm name="Ort">
      <wert>Bremen</wert>
    </mm>
  </mmgr>
</mmgr>
<!-- beteiligtePerson -->
</mmgr>
<!-- gewerbe -->
<mm name="GueltigAb">
  <wert>01012003</wert>
</mm>
<mm name="DatumMeldung">
  <wert>17012003</wert>
</mm>
<mmgr name="Taetigkeiten">
  <mm name="Taetigkeit">
    <wert>Einzelhandel mit Lebensmitteln und Gebrauchsgütern aller Art, Spirituosen,
Tabak- und Drogeriewaren</wert>
  </mm>
  <mm name="TaetigkeitSchluessel">
    <wert>52112</wert>
  </mm>
  <mm name="WZGSchluesselVerzeichnis">
    <wert>WZG012003</wert>
  </mm>
  <mm name="IstSchwerpunkt">
    <wert>1</wert>
  </mm>
</mmgr>
<mm name="GrundAnmeldungSchluessel">

```

```
    <wert>03</wert>
  </mm>
</satz>
</datensegment>
</nachricht>
</DatML-RAW-D>
```

6. DTD und XML-Schema 2.0 [nicht-normativ]

Dieser Ausdruck der DTD kann nicht als Referenz verwendet werden.

6.1. DTD

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- *****
Schema      : DatML/RAW
Version     : 2.0
Variante    : D
Edition     : 0
Typ         : XML-DTD
Namespace   : http://www.destatis.de/schema/datml-raw/2.0/de
Stand       : 20041104
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:

DatML/RAW ist ein Dokumenttyp fuer die Lieferung statistischer Daten
durch Berichtspflichtige oder in deren Namen handelnde Dritte an die
Statistischen Aemter oder fuer diese empfangende Stellen.

Mit DatML/RAW koennen Daten fuer beliebige Erhebungen und Berichtszeit-
raeume als Erst-, Nach- und Korrekturlieferung uebertragen werden.

Eine ausfuehrliche Spezifikation ist vom Autor und unter der Internet-
Adresse http://www.statspez.de erhaeltlich.

Updates (U) und Korrekturen (K):

*** Version 1.0.0 ***

001 K 20021020/K/1.0.0/D//Michael Schaefer
      - Element <mmgr> in Inhaltsmodell von <satz> aufgenommen

*** Version 1.0.2 ***

002 U 20030516/U/1.0.2/D//Michael Schaefer
      - Alle Versionsbezeichner der kompatiblen Vorgaengerversionen
        sind im Attribut version zulaessig.
      - Element <identifikation> als optional erkluert.
      - Neues Element <url> als Kindknoten von <kontakt>
      - Neue Elemente <postfachleitzahl> und <postfachort>
      - <telefon>, <email> und <fax> sowie <url> duerfen nur noch
        einmal und in dieser Reihenfolge angegeben werden, jedoch sind
        alle Elemente weiterhin optional.

*** Version 2.0.0 ***

Version 2.0.0 ist in folgenden Punkten inkompatibel zu 1.0.2:
- <eingang> ist ersetzt durch <dateneingang>
- <instanz> ist ersetzt durch <dokumentinstanz>
- feste Reihenfolge der Metadaten in <segment> und <nachricht>
- <lieferattribute> ist ersetzt durch <meldungsart>
- Kapselung der Meldungsdaten mit <datensegment>
- Geaenderte Attribute in <wert>

003 U 20041028/U/2.0.0/D//Michael Schaefer
      - Nachricht/Segment:
        - Kapselung der Datensatze im Datensegment mit <datensegment>
        - Neues Element <nachrichtenID>
        - Feste Reihenfolge Metadaten in <segment> und <nachricht>

      - Absender und Empfaenger
        - Neues Element <externeIdentifikation>
```

```

- Identifikation und Kontaktangaben
  - <telefon>, <email>, <fax>, <url> in <identifikation>
  - <adresse> in <kontakt>
  - Neues Element <hausnummer>

- Statistischer Kontext
  - <lieferattribute> ersetzt durch <meldungsart>
  - Neues Element <externeIdentifikation>
  - Neues Element <semester>

- Protokollierung:
  - Neues Element <dateneingang> ersetzt <eingang>
  - Neues Element <dokumentinstanz> ersetzt <instanz>
  - Neue Elemente <anwendung>, <anwendungsname>, <dateiname>,
    <datenformat>, <datentraeger>, <dokumentID>, <datei>,
    <eingangsstempel>, <formularID>, <formularname>,
    <hersteller>, <onlineformular>, <papierformular>,
    <ressourceID>, <server>,<upload>, <zertifikat>
  - Neue Kindelemente <datum> und <uhrzeit>

- Daten:
  - Geschachtelte Merkmalsgruppen zugelassen:
    - Element <mmgr> darf sich selbst enthalten
  - Kapselung der Datensaeetze im Datensegment mit <datensegment>
  - Neue Elemente <meldungsID>, <meldungsart> mit nachgeordneten
    Elementen <original>, <berichtigung>, <nachmeldung>, <austausch>,
    <loeschung>, <nil>
  - Bisherige Attribute in <wert> entfallen, neue Attribute
    einheit, klasse und faktor ohne Vorbelegung (ausser faktor).

- Sonstige Elemente:
  - Neue Elemente <datum> und <uhrzeit>

- Sonstiges:
  - Datumsklasse "format" um Semesterangaben erweitert (Details
    in der Spezifikation).

*****
-->

<!-- Root-Element
Das Attribut 'version' muss immer angegeben werden. Absender, Empfaenger
und mindestens eine Nachricht sind Mindestbestandteile eines DatML/RAW-
Dokumentes.

2.0.0 : Attribut Version mit neuer Werteliste
-->
<!ELEMENT DatML-RAW-D (memo?, optionen?, protokoll?, absender, empfaenger, nachricht+,
anzahl?)>
<!ATTLIST DatML-RAW-D
    version (2.0) #REQUIRED
>

<!-- Nachricht, Segment
-->

<!-- nachricht
Eine Nachricht enthaelt ein oder mehr Meldungen; im letzten Fall muss
die Nachricht segmentiert sein.

2.0.0 : Neue Kindelemente <nachrichtenID> und <datensegment>, lieferattribute ent-
faellt,
    feste Reihenfolge der Metadaten
-->
<!ELEMENT nachricht (memo?, nachrichtenID?, erhebung?, berichtszeitraum?, berichtspflich-
tiger?, berichtsempfaenger?, material?, datenattribute?, hmm*, (datensegment | segment+))>

<!-- nachrichtenID
Eine (eindeutige) ID fuer die Nachricht. Eine vom Absender vergebene
ID soll erhalten bleiben, ansonsten ist die Vergabe optional.

2.0.0 : Neues Element
-->
<!ELEMENT nachrichtenID (#PCDATA)>
<!ATTLIST nachrichtenID
    klasse CDATA #IMPLIED

```

```

>

<!-- segment
  Segmente dienen der hierarchischen Anordnung von Metadaten. Segmente
  koennen weitere Segmente enthalten, wodurch ein sogenannter Segment-
  baum entsteht.

  2.0.0 : Neues Kindelement <datensegment>, <lieferattribute> entfaellt,
  feste Reihenfolge der Metadaten
  -->
  <!ELEMENT segment (memo?, erhebung?, berichtszeitraum?, berichtspflichtiger?, bericht-
  sempfaenger?, material?, datenattribute?, hmm*, (datensegment | segment+))>

<!-- Absender und Empfaenger
  -->

  <!-- absender
    1.0.2 : Neues Kindelement <externeIdentifikation>
    -->
    <!ELEMENT absender (kennung, berechtigung?, identifikation?, externeIdentifikation?, kon-
    takt?, korrektur?, memo?)>

  <!-- empfaenger
    1.0.2 : Neues Kindelement <externeIdentifikation>
    -->
    <!ELEMENT empfaenger (kennung?, identifikation?, externeIdentifikation?, kontakt?, me-
    mo?)>

<!-- Identifikation und Kontaktangaben
  -->
  <!ELEMENT adresse (strasse?, hausnummer?, postfach?, postfachleitzahl?, postfachort?,
  postleitzahl, ort, kreis?, bundesland?, land?, zusatz?)>
  <!ELEMENT anrede (#PCDATA)>
  <!ATTLIST anrede
    zeilennummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
  >

  <!ELEMENT bundesland (#PCDATA)>
  <!ATTLIST bundesland
    zeilennummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
  >

  <!ELEMENT email (#PCDATA)>
  <!ATTLIST email
    zeilennummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
  >

  <!-- externeIdentifikation
    2.0.0 : Neues Element
    -->
  <!ELEMENT externeIdentifikation (#PCDATA)>
  <!ATTLIST externeIdentifikation
    klasse CDATA #IMPLIED
  >

  <!ELEMENT fax (#PCDATA)>
  <!ATTLIST fax zeilennummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
  >

  <!-- hausnummer
    2.0.0 : Neues Element
    -->
  <!ELEMENT hausnummer (#PCDATA)>
  <!ATTLIST hausnummer zeilennummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
  >

  <!-- identifikation
    2.0.0 : Neue Kindelemente telefon, email, fax, url
    -->
  <!ELEMENT identifikation (zeile+ | (identitaet, adresse?, telefon?, email?, fax?, url?))>

  <!-- identitaet

```

```
-->
<!ELEMENT identitaet (organisation?, person?)>

<!-- kontakt
    1.0.2 : feste Reihenfolge telefon, email, fax, neues Kindelement url
    2.0.0 : Neues Kindelement adresse
-->
<!ELEMENT kontakt (zeile+ | (identitaet?, adresse?, telefon?, email?, fax?, url?))>

<!ELEMENT korrektur (identifikation?, kontakt?)>

<!ELEMENT kreis (#PCDATA)>
<!ATTLIST kreis
    zeilennummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
>

<!ELEMENT land (#PCDATA)>
<!ATTLIST land
    zeilennummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
>

<!ELEMENT nachname (#PCDATA)>
<!ATTLIST nachname
    zeilennummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
>

<!ELEMENT niederlassung (#PCDATA)>
<!ATTLIST niederlassung
    zeilennummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
>

<!ELEMENT organisation (name, niederlassung?, zusatz?)>
<!ELEMENT ort (#PCDATA)>
<!ATTLIST ort zeilennummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
>

<!ELEMENT person (anrede?, titel?, vorname?, nachname, zusatz?)>
<!ELEMENT postfach (#PCDATA)>
<!ATTLIST postfach
    zeilennummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
>

<!ELEMENT postfachleitzahl (#PCDATA)>
<!ATTLIST postfachleitzahl
    zeilennummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
>

<!ELEMENT postfachort (#PCDATA)>
<!ATTLIST postfachort
    zeilennummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
>

<!ELEMENT postleitzahl (#PCDATA)>
<!ATTLIST postleitzahl
    zeilennummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
>

<!ELEMENT strasse (#PCDATA)>
<!ATTLIST strasse
    zeilennummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
>

<!ELEMENT telefon (#PCDATA)>
<!ATTLIST telefon
    zeilennummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
>
```

```

<!ELEMENT titel (#PCDATA)>
<!ATTLIST titel
    zeilenummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
    >

<!-- hausnummer
    1.0.2 : Neues Element
-->
<!ELEMENT url (#PCDATA)>
<!ATTLIST url zeilenummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
    >

<!ELEMENT vorname (#PCDATA)>
<!ATTLIST vorname
    zeilenummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
    >

<!ELEMENT zeile (#PCDATA)>
<!ATTLIST zeile
    zeilenummer CDATA #IMPLIED
    html-name CDATA #IMPLIED
    >

<!-- Statistischer Kontext
-->

<!-- beginn
    2.0.0 : Neues Kindelement semester
-->
<!ELEMENT beginn (jahr, (halbjahr | semester | quartal | (monat, tag?) | woche?))>

<!-- berichtsempfaenger
    1.0.2 : <identifikation> optional, neues Kindelement <externeIdentifikation>
-->
<!ELEMENT berichtsempfaenger (kennung?, identifikation?, externeIdentifikation?, kon-
takt?, memo?)>

<!-- berichtspflichtiger
    1.0.2 : <identifikation> optional, neues Kindelement <externeIdentifikation>
-->
<!ELEMENT berichtspflichtiger (kennung, berechtigung?, identifikation?, externeIdentifi-
kation?, kontakt?, korrektur?, memo?)>

<!-- berichtszeitraum
    2.0.0 : Neues Kindelement semester
-->
<!ELEMENT berichtszeitraum (string | (jahr, (halbjahr | semester | quartal | (monat,
tag?) | woche?)) | (beginn, ende))>

<!-- datenattribute
-->
<!ELEMENT datenattribute EMPTY>
<!ATTLIST datenattribute
    dezimalzeichen CDATA ","
    tausender-trennzeichen CDATA "."
    >

<!-- ende
    2.0.0 : Neues Kindelement semester
-->
<!ELEMENT ende (jahr, (halbjahr | semester | quartal | (monat, tag?) | woche?))>

<!-- erhebung
-->
<!ELEMENT erhebung (kennung, text?, berechtigung?)>

<!-- halbjahr
-->
<!ELEMENT halbjahr (#PCDATA)>

<!-- jahr
-->
<!ELEMENT jahr (#PCDATA)>

```

```

<!-- lieferattribute
      2.0.0 : Element entfallen
-->

<!-- material
-->
<!ELEMENT material (name?, kennung?, version?)>

<!-- monat
-->
<!ELEMENT monat (#PCDATA)>

<!-- quartal
-->
<!ELEMENT quartal (#PCDATA)>

<!-- semester
      2.0.0 : Neues Element
-->
<!ELEMENT semester (#PCDATA)>

<!-- tag
-->
<!ELEMENT tag (#PCDATA)>

<!-- woche
-->
<!ELEMENT woche (#PCDATA)>

<!-- Daten
-->

<!-- austausch
      2.0.0 : Neues Element
-->
<!ELEMENT austausch (meldungsID)>

<!-- berichtigung
      2.0.0 : Neues Element
-->
<!ELEMENT berichtigung (meldungsID)>

<!-- datensegment
      Das Element datensegment stellt einen Blattknoten im Segmentbaum dar. Ab
      einem Element datensegment ist der Kontext einer Meldung vollstaendig be-
      schrieben und die Meldung ist eindeutig. Daher werden Meldungs-ID
      und Meldungsart hier abgelegt.

      2.0.0 : Neues Element
-->
<!ELEMENT datensegment (meldungsID?, meldungsart?, (omm | satz+))>

<!-- hmm
-->
<!ELEMENT hmm (memo?, wert)>
<!ATTLIST hmm name CDATA #REQUIRED
              text CDATA #IMPLIED
              klasse CDATA #IMPLIED
              html-name CDATA #IMPLIED
              >

<!-- loeschung
      2.0.0 : Neues Element
-->
<!ELEMENT loeschung (meldungsID)>

<!-- meldungsart
      Die Meldungsart wird durch ein nachgeordnetes Element ausgedrueckt.
      Einige Funktionen erfordern den Bezug zu einer vorausgegangenen Meldung,
      z.B. Berichtigungen. In diesem Fall enthalten sie ein weiteres Element
      meldungsID mit der Meldungs-ID.

      2.0.0 : Neues Element
-->

<!-- meldungsID
      2.0.0 : Neues Element

```

```

    -->
<!ELEMENT meldungsID (#PCDATA)>
<!ATTLIST meldungsID
    klasse CDATA #IMPLIED
>

<!ELEMENT meldungsart (original|berichtigung|nachmeldung|austausch|loeschung|nil)>

<!-- mm
    -->
<!ELEMENT mm (memo?, (wert | na))>
<!ATTLIST mm name CDATA #REQUIRED
    text CDATA #IMPLIED
    klasse CDATA #IMPLIED
    html-name CDATA #IMPLIED
>

<!-- mmgr
    1.0.2 . Neues Element
    -->
<!ELEMENT mmgr ((mm | mmgr)+)>
<!ATTLIST mmgr
    name CDATA #REQUIRED
    klasse CDATA #IMPLIED
    text CDATA #IMPLIED
    html-name CDATA #IMPLIED
    index CDATA #IMPLIED
>

<!-- na
    -->
<!ELEMENT na (#PCDATA)>

<!-- nachmeldung
    2.0.0 : Neues Element
    -->
<!ELEMENT nachmeldung (meldungsID?)>

<!-- nil
    2.0.0 : Neues Element
    -->
<!ELEMENT nil EMPTY>

<!-- omm
    -->
<!ELEMENT omm (memo?, (wert, (omm | satz+)))+>
<!ATTLIST omm name CDATA #REQUIRED
    text CDATA #IMPLIED
    klasse CDATA #IMPLIED
    html-name CDATA #IMPLIED
>

<!-- original
    2.0.0 : Neues Element
    -->
<!ELEMENT original EMPTY>

<!-- satz
    -->
<!ELEMENT satz ((mm | mmgr)+, anzahl?)>
<!ATTLIST satz
    kennung CDATA #IMPLIED
>

<!-- wert
    -->
<!ELEMENT wert (#PCDATA)>
<!ATTLIST wert
    einheit CDATA #IMPLIED
    faktor CDATA "1"
    klasse CDATA #IMPLIED
>

<!-- Verarbeitungssteuerung
Es ist moeglich, Testdokumente zu kennzeichnen und eine Empfangsbestae-
tigung anzufordern (Unterstuetzung vom Implementierungsgrad und ggf.

```

```

von der Erhebung abhaengig)
-->

<!-- optionen
-->
<!ELEMENT optionen (test?, empfangsbestaetigung?)>

<!-- test
-->
<!ELEMENT test EMPTY>
<!ATTLIST test
      kennung CDATA #IMPLIED
    >

<!-- empfangsbestaetigung
-->
<!ELEMENT empfangsbestaetigung EMPTY>
<!ATTLIST empfangsbestaetigung
      versandart (default | email | post | download ) "default"
      format (default | xml | text) "default"
      betreff CDATA #IMPLIED
    >

<!-- Protokollierung
Der Dateneingang und beliebig viele Dokumentinstanzen koennen proto-
kolliert werden. Eine Dokumentinstanz entspricht einer Instanz von
DatML/RAW. Lieferweg und Datenquelle werden im Element <dateneingang>
beschrieben.
Die Elemente <eingang> und <instanz> werden ab Version 2.0 durch
<dateneingang> bzw. <dokumentinstanz> ersetzt.
-->

<!-- anwendung
      2.0.0 : Neues Element
-->
<!ELEMENT anwendung (anwendungsname, version?, server?, hersteller?, zertifikat?)>

<!-- anwendungsname
      2.0.0 : Neues Element
-->
<!ELEMENT anwendungsname (#PCDATA)>

<!-- datei
      2.0.0 : Neues Element
-->
<!ELEMENT datei (dateiname?, datum?, uhrzeit?, anwendung?, dokumentID?, datenformat?,
(datentraeger|upload)?)>

<!-- dateiname
      2.0.0 : Neues Element
-->
<!ELEMENT dateiname (#PCDATA)>

<!-- datenformat
      2.0.0 : Neues Element
-->
<!ELEMENT datenformat (#PCDATA)>
<!ATTLIST datenformat
      typ (undefiniert | datml | xml | datensatz | csv | edifact) "undefi-
niert"
    >

<!-- datentraeger
      2.0.0 : Neues Element
-->
<!ELEMENT datentraeger (#PCDATA)>
<!ATTLIST datentraeger
      typ (undefiniert | diskette | cdrom | dvd | kassette | magnetband)
"undefiniert"
    >

<!-- dateneingang
      2.0.0 : Neues Element
-->
<!ELEMENT dateneingang (datum, uhrzeit, eingangsstempel, anwendung, (papierformu-
lar|onlineformular|datei)?)>

```

```
<!-- dokumentID
  2.0.0 : Neues Element
-->
<!ELEMENT dokumentID (#PCDATA)>
<ATTLIST dokumentID
  klasse CDATA #IMPLIED
>

<!-- ressourceID
  2.0.0 : Neues Element
-->
<!ELEMENT ressourceID (#PCDATA)>
<ATTLIST ressourceID
  klasse CDATA #IMPLIED
>

<!-- dokumentinstanz
  2.0.0 : Neues Element
-->
<!ELEMENT dokumentinstanz (dateiname?, datum, uhrzeit, anwendung?, dokumentID?, ressourceID*)>

<!-- eingang
  2.0.0 : Element entfallen
-->

<!-- eingangsstempel
  2.0.0 : Neues Element
-->
<!ELEMENT eingangsstempel (#PCDATA)>
<ATTLIST eingangsstempel
  klasse CDATA #IMPLIED
>

<!-- formularID
  2.0.0 : Neues Element
-->
<!ELEMENT formularID (#PCDATA)>
<ATTLIST formularID
  klasse CDATA #IMPLIED
>

<!-- formularname
  2.0.0 : Neues Element
-->
<!ELEMENT formularname (#PCDATA)>
<ATTLIST formularname
  klasse CDATA #IMPLIED
>

<!-- hersteller
  2.0.0 : Neues Element
-->
<!ELEMENT hersteller (#PCDATA)>

<!-- instanz
  2.0.0 : Element entfallen
-->

<!-- onlineformular
  2.0.0 : Neues Element
-->
<!ELEMENT onlineformular (formularID?, formularname?)>

<!-- papierformular
  2.0.0 : Neues Element
-->
<!ELEMENT papierformular (formularID?, formularname?)>

<!-- protokoll
  2.0.0 : Elemente eingang und instanz ersetzt durch dateneingang bzw.
  dokumentinstanz
-->
<!ELEMENT protokoll (dateneingang?, dokumentinstanz*)>

<!-- upload
  2.0.0 : Neues Element
-->
```

```
<!ELEMENT upload EMPTY>

<!-- server
      2.0.0 : Neues Element
-->
<!ELEMENT server (#PCDATA)>

<!-- zertifikat
      2.0.0 : Neues Element
-->
<!ELEMENT zertifikat (#PCDATA)>
<!ATTLIST zertifikat
      klasse CDATA #IMPLIED
>

<!-- Sonstige Elemente
-->

<!-- anzahl
      -->
<!ELEMENT anzahl (#PCDATA)>

<!-- berechtigung
      -->
<!ELEMENT berechtigung (#PCDATA)>
<!ATTLIST berechtigung
      klasse CDATA "default"
>

<!-- datum
      2.0.0 : Neues Element
-->
<!ELEMENT datum (#PCDATA)>
<!ATTLIST datum
      format CDATA #FIXED "jjjjmmtt"
      klasse CDATA #FIXED "datum"
>

<!-- kennung
      -->
<!ELEMENT kennung (#PCDATA)>
<!ATTLIST kennung
      klasse CDATA "default"
>

<!-- memo
      -->
<!ELEMENT memo (#PCDATA)>

<!-- name
      -->
<!ELEMENT name (#PCDATA)>
<!ATTLIST name
      zeilennummer CDATA #IMPLIED
      html-name CDATA #IMPLIED
>

<!-- string
      -->
<!ELEMENT string (#PCDATA)>
<!ATTLIST string
      format CDATA #REQUIRED
      klasse (undefiniert | datum) #REQUIRED
>

<!-- text
      -->
<!ELEMENT text (#PCDATA)>

<!-- uhrzeit
      2.0.0 : Neues Element
-->
<!ELEMENT uhrzeit (#PCDATA)>
<!ATTLIST uhrzeit
      format CDATA #FIXED "hmmss"
      klasse CDATA #FIXED "datum"
>
```

```

<!-- version
-->
<!ELEMENT version (#PCDATA)>

<!-- zusatz
2.0.0 : Neue Attribute zeilennummer und html-name
-->
<!ELEMENT zusatz (#PCDATA)>
<!ATTLIST zusatz
        zeilennummer CDATA #IMPLIED
        html-name CDATA #IMPLIED
        >

<!-- Ende der DTD -->

```

6.2. XML-Schema

```

<?xml version = "1.0" encoding = "ISO-8859-1"?>
<xs:schema xmlns = "http://www.destatis.de/schema/datml-raw/2.0/de"
  targetNamespace = "http://www.destatis.de/schema/datml-raw/2.0/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">
    <![CDATA[
      Schema      : DatML/RAW
      Version     : 2.0
      Variante    : D
      Edition     : 0
      Revision    : 1
      Typ         : XML-Schema
      Namensraum : http://www.destatis.de/schema/datml-raw/2.0/de
      Stand       : 20050110
      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:

DatML/RAW ist ein Dokumenttyp für die Lieferung statistischer Daten durch Berichtspflichtige oder in deren Namen handelnde Dritte an die Statistischen Ämter oder für diese empfangende Stellen.

Mit DatML/RAW können Daten für beliebige Erhebungen und Berichtszeitraume als Erst-, Nach- und Korrekturlieferung übertragen werden.

Eine ausführliche Spezifikation ist unter der Internet-Adresse <http://www.statspez.de> erhältlich.

Updates (U), Korrekturen (K), Revisionen (R):

*** Version 1.0.0 ***

001 K 20021020/K/1.0.0/D//Michael Schaefer
- Element <mmgr> in Inhaltsmodell von <satz> aufgenommen

*** Version 1.0.2 ***

002 U 20030516/U/1.0.2/D//Michael Schaefer
- Alle Versionsbezeichner der kompatiblen Vorgängerversionen sind im Attribut version zulaessig.
- Element <identifikation> als optional erklart.
- Neues Element <url> als Kindknoten von <kontakt>
- Neue Elemente <postfachleitzahl> und <postfachort>
- <telefon>, <email> und <fax> sowie <url> duerfen nur noch einmal und in dieser Reihenfolge angegeben werden, jedoch sind alle Elemente weiterhin optional.

*** Version 2.0.0 ***

Version 2.0.0 ist in folgenden Punkten inkompatibel zu 1.0.2:

- <ingang> ist ersetzt durch <dateneingang>
- <instanz> ist ersetzt durch <dokumentinstanz>
- feste Reihenfolge der Metadaten in <segment> und <nachricht>
- <lieferattribute> ist ersetzt durch <meldungsart>
- Kapselung der Meldungsdaten mit <datensegment>
- Geaenderte Attribute in <wert>

003 U 20041028/U/2.0.0/D//Michael Schaefer

- Nachricht/Segment:
 - Kapselung der Datensatze im Datensegment mit <datensegment>
 - Neues Element <nachrichtenID>
 - Feste Reihenfolge Metadaten in <segment> und <nachricht>
 - Abender und Empfaenger
 - Neues Element <externeIdentifikation>
- Identifikation und Kontaktangaben
 - <telfon>, <email>, <fax>, <url> in <identifikation>
 - <adrese> in <kontakt>
 - Neues lement <hausnummer>
- Statistischer Kontext
 - <lieferattribute> ersetzt durch <meldungsart>
 - Neues Elemnt <externeIdentifikation>
 - Neues Elemet <semester>
- Protokollierung
 - Neues Element <dateneingang> ersetzt <ingang>
 - Neues Element <okumentinstanz> ersetzt <instanz>
 - Neue Elemente <awendung>, <anwendungsname>, <dateiname>, <datenformat>, <dtentraeger>, <dokumentID>, <datei>, <eingangsstempel>, <formularID>, <formularname>, <hersteller>, <onlineformular>, <papierformular>, <ressourceID>, <server>, <upload>, <zertifikat>
 - Neue Kindelemente <daum> und <uhrzeit>
- Daten:
 - Gehachtelte Merkmalsgrppen zugelassen:
 - Element <mmgr> darf sic selbst enthalten
 - Kapselung der Datensatze m Datensegment mit <datensegment>
 - Neue Elemente <meldungsID>, <meldungsart> mit nachgeordneten Elementen <original>, <berichtigung>, <nachmeldung>, <austausch>, <loeschung>, <nil>
 - Bisherige Attribute in <wert> etfallen, neue Attribute einheit, klasse und faktor ohne orbelegung (ausser faktor).
- Sonstige Elemente:
 - Neue Elemente <datum> und <uhrzeit>
- Sonstiges:
 - Datumsklasse "format" um Semensterangaen erweitert (Details in der Spezifikation).

003 R 20050110/R/2.0.0/D/1/Michael Schaefer

- Anpassung an Spezifikation:
 - Element <korrektur> in <meldungsart> in <austausch> umbenannt.
 - Typ IdentifikationKontaktTyp durch separate Typen IdentifikationTyp und KontaktTyp ersetzt.

]]>

```

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

<!-- Allgemeine Typen -->

<xs:complexType name = "DatumTyp">
  <xs:simpleContent>
    <xs:extension base = "xs:string">
      <xs:attribute name = "format" fixed = "jjjmmmtt" 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 = "TextZeileHTMLTyp">
    <xs:simpleContent>
        <xs:extension base = "xs:string">
            <xs:attributeGroup ref = "ZeileHTMLAttrGruppe"/>
        </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: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:element name = "empfangsbestaetigung" type="EmpfangsbestaetigungTyp" minOccurs =
"0" />
    </xs:sequence>
</xs:complexType>

<xs:complexType name="EmpfangsbestaetigungTyp">
    <xs:attribute name = "versandart" default = "default">
        <xs:simpleType>
            <xs:restriction base = "xs:token">
                <xs:enumeration value = "default"/>
                <xs:enumeration value = "email"/>
                <xs:enumeration value = "post"/>
                <xs:enumeration value = "download"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
    <xs:attribute name = "format" default = "default">
        <xs:simpleType>
            <xs:restriction base = "xs:token">
                <xs:enumeration value = "default"/>
                <xs:enumeration value = "xml"/>
                <xs:enumeration value = "text"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
    <xs:attribute name = "betreff" type = "xs:normalizedString"/>
</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 = "ZusatzTyp">
  <xs:simpleContent>
    <xs:extension base = "TextZeileHTMLTyp"/>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name = "ProtokollTyp">
  <xs:sequence>
    <xs:element name = "dateneingang" type = "DateneingangTyp" minOccurs = "0"/>
    <xs:element name = "dokumentinstanz" type = "DokumentinstanzTyp" minOccurs = "0" maxOc-
curs = "unbounded"/>
  </xs:sequence>
</xs:complexType>

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

<!-- Veraltet; ersetzt durch IdentifikationTyp ud KontaktTyp
<xs:complexType name = "IdentifikationKontaktTyp">
  <xs:choice>
    <xs:element name = "zeile" type = "TextZeileHTMLTyp" maxOccurs = "unbounded"/>
    <xs:sequence>
      <xs:element name = "identitaet" type = "IdentitaetTyp" minOccurs = "0"/>
      <xs:element name = "adresse" type = "AdresseTyp" minOccurs = "0"/>
      <xs:element name = "telefon" type = "TextZeileHTMLTyp" minOccurs = "0"/>
      <xs:element name = "email" type = "TextZeileHTMLTyp" minOccurs = "0"/>
      <xs:element name = "fax" type = "TextZeileHTMLTyp" minOccurs = "0"/>
      <xs:element name = "url" type = "TextZeileHTMLTyp" minOccurs = "0"/>
    </xs:sequence>
  </xs:choice>
</xs:complexType>
-->

<xs:complexType name = "IdentifikationTyp">
  <xs:choice>
    <xs:element name = "zeile" type = "TextZeileHTMLTyp" maxOccurs = "unbounded"/>
    <xs:sequence>
      <xs:element name = "identitaet" type = "IdentitaetTyp"/>
      <xs:element name = "adresse" type = "AdresseTyp" minOccurs = "0"/>
      <xs:element name = "telefon" type = "TextZeileHTMLTyp" minOccurs = "0"/>
      <xs:element name = "email" type = "TextZeileHTMLTyp" minOccurs = "0"/>
      <xs:element name = "fax" type = "TextZeileHTMLTyp" minOccurs = "0"/>
      <xs:element name = "url" type = "TextZeileHTMLTyp" minOccurs = "0"/>
    </xs:sequence>
  </xs:choice>
</xs:complexType>

```

```

<xs:complexType name = "KontaktTyp">
  <xs:choice>
    <xs:element name = "zeile" type = "TextZeileHTMLTyp" maxOccurs = "unbounded"/>
    <xs:sequence>
      <xs:element name = "identitaet" type = "IdentitaetTyp" minOccurs = "0"/>
      <xs:element name = "adresse" type = "AdresseTyp" minOccurs = "0"/>
      <xs:element name = "telefon" type = "TextZeileHTMLTyp" minOccurs = "0"/>
      <xs:element name = "email" type = "TextZeileHTMLTyp" minOccurs = "0"/>
      <xs:element name = "fax" type = "TextZeileHTMLTyp" minOccurs = "0"/>
      <xs:element name = "url" type = "TextZeileHTMLTyp" 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 = "TextZeileHTMLTyp"/>
    <xs:element name = "niederlassung" type = "TextZeileHTMLTyp" minOccurs = "0"/>
    <xs:element name = "zusatz" type = "ZusatzTyp" minOccurs = "0"/>
  </xs:sequence>
</xs:complexType>

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

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

<xs:complexType name = "KorrekturTyp">
  <xs:sequence>
    <xs:element name = "identifikation" type = "IdentifikationTyp" minOccurs = "0"/>
    <xs:element name = "kontakt" type = "KontaktTyp" 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:complexType>

```

```

        <xs:element name = "beginn" type = "BeginnEndeTyp"/>
        <xs:element name = "ende" type = "BeginnEndeTyp"/>
    </xs:sequence>
</xs:choice>
</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>

<!-- Typen Datenattribute und Material -->

```

```

<xs:complexType name = "DatenattributeTyp">
  <xs:attribute name = "dezimalzeichen" default = "," type = "xs:normalizedString"/>
  <xs:attribute name = "tausender-trennzeichen" default = "." type = "xs:normalizedString"/>
</xs:complexType>

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

<!-- Typen Merkmale und Datensätze -->

<xs:complexType name = "HMMTyp">
  <xs:sequence>
    <xs:element name = "memo" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "wert" type = "WertTyp"/>
  </xs:sequence>
  <xs:attributeGroup ref = "NameTextKlasseHTMLAttrGruppe"/>
</xs:complexType>

<xs:complexType name = "WertTyp">
  <xs:simpleContent>
    <xs:extension base = "xs:string">
      <xs:attribute name = "einheit" type = "xs:normalizedString"/>
      <xs:attribute name = "faktor" default = "1" type = "xs:positiveInteger"/>
      <xs:attribute name = "klasse" type = "xs:normalizedString"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name = "OMMTyp">
  <xs:sequence>
    <xs:element name = "memo" type = "xs:string" minOccurs = "0"/>
    <xs:sequence maxOccurs = "unbounded">
      <xs:element name = "wert" type = "WertTyp"/>
      <xs:choice>
        <xs:element name = "omm" type = "OMMTyp"/>
        <xs:element name = "satz" type = "SatzTyp" maxOccurs = "unbounded"/>
      </xs:choice>
    </xs:sequence>
  </xs:sequence>
  <xs:attributeGroup ref = "NameTextKlasseHTMLAttrGruppe"/>
</xs:complexType>

<xs:complexType name = "MMTyp">
  <xs:sequence>
    <xs:element name = "memo" type = "xs:string" minOccurs = "0"/>
    <xs:choice>
      <xs:element name = "wert" type = "WertTyp"/>
      <xs:element name = "na" type = "xs:normalizedString"/>
    </xs:choice>
  </xs:sequence>
  <xs:attributeGroup ref = "NameTextKlasseHTMLAttrGruppe"/>
</xs:complexType>

<xs:complexType name = "MMGRTyp">
  <xs:choice maxOccurs = "unbounded">
    <xs:element name = "mm" type = "MMTyp"/>
    <xs:element name = "mmgr" type = "MMGRTyp"/>
  </xs:choice>
  <xs:attributeGroup ref = "NameTextKlasseHTMLAttrGruppe"/>
  <xs:attribute name = "index" type = "xs:normalizedString"/>
</xs:complexType>

<xs:complexType name = "SatzTyp">
  <xs:sequence>
    <xs:choice maxOccurs = "unbounded">
      <xs:element name = "mm" type = "MMTyp"/>
      <xs:element name = "mmgr" type = "MMGRTyp"/>
    </xs:choice>
    <xs:element name = "anzahl" type = "xs:positiveInteger" minOccurs = "0"/>
  </xs:sequence>
  <xs:attribute name = "kennung" type = "xs:normalizedString"/>
</xs:complexType>

```

```

<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 Nachricht, Segment, Datensegment -->

<xs:complexType name = "NachrichtTyp">
  <xs:sequence>
    <xs:element name = "memo" type = "xs:string" minOccurs = "0"/>
    <xs:element name = "nachrichtenID" type = "NachrichtenIDTyp" minOccurs = "0"/>
    <xs:group ref = "NachrichtSegmentGruppe"/>
  </xs:sequence>
</xs:complexType>

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

<xs:complexType name = "DatensegmentTyp">
  <xs:sequence>
    <xs:element name = "meldungsID" type = "MeldungsIDTyp" minOccurs = "0"/>
    <xs:element name = "meldungsart" type = "MeldungsartTyp" minOccurs = "0"/>
    <xs:choice>
      <xs:element name = "omm" type = "OMMTyp"/>
      <xs:element name = "satz" type = "SatzTyp" maxOccurs = "unbounded"/>
    </xs:choice>
  </xs:sequence>
</xs:complexType>

<xs:complexType name = "SegmentTyp">
  <xs:sequence>
    <xs:element name = "memo" type = "xs:string" minOccurs = "0"/>
    <xs:group ref = "NachrichtSegmentGruppe"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name = "AbsenderBerichtspflichtigerTyp">
  <xs:sequence>
    <xs:element name = "kennung" type = "TextKlasseTyp"/>
    <xs:element name = "berechtigung" type = "TextKlasseTyp" minOccurs = "0"/>
    <xs:element name = "identifikation" type = "IdentifikationTyp" minOccurs = "0"/>
    <xs:element name = "externeIdentifikation" type = "TextKlasseTyp" minOccurs = "0"/>
    <xs:element name = "kontakt" type = "KontaktTyp" minOccurs = "0"/>
    <xs:element name = "korrektur" type = "KorrekturTyp" minOccurs = "0"/>
    <xs:element name = "memo" type = "xs:string" minOccurs = "0"/>
  </xs:sequence>
</xs:complexType>

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

```

```

</xs:complexType>

<!-- Convenience Type -->

<xs:complexType name = "KontextType">
  <xs:sequence>
    <xs:group ref="KontextGruppe"/>
  </xs:sequence>
</xs:complexType>

<!-- Gruppen -->

<xs:group name = "BerichtszeitraumGruppe">
  <xs:sequence>
    <xs:element name = "jahr" type="JahrTyp"/>
    <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:sequence>
        <xs:element name = "monat" type = "MonatTyp"/>
        <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 = "0"/>
    <xs:element ref = "berichtszeitraum" minOccurs = "0"/>
    <xs:element ref = "berichtspflichtiger" minOccurs = "0"/>
    <xs:element ref = "berichtsempfaenger" minOccurs = "0"/>
    <xs:element name = "hmm" type = "HMMTyp" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:group>

<xs:group name = "NachrichtSegmentGruppe">
  <xs:sequence>
    <xs:element ref = "erhebung" minOccurs = "0"/>
    <xs:element ref = "berichtszeitraum" minOccurs = "0"/>
    <xs:element ref = "berichtspflichtiger" minOccurs = "0"/>
    <xs:element ref = "berichtsempfaenger" minOccurs = "0"/>
    <xs:element name = "material" type = "MaterialTyp" minOccurs = "0"/>
    <xs:element name = "datenattribute" type = "DatenattributeTyp" minOccurs = "0"/>
    <xs:element name = "hmm" type = "HMMTyp" minOccurs="0" maxOccurs="unbounded"/>
    <xs:choice>
      <xs:element ref = "datensegment"/>
      <xs:element ref = "segment" maxOccurs = "unbounded"/>
    </xs:choice>
  </xs:sequence>
</xs:group>

<xs:attributeGroup name = "ZeileHTMLAttrGruppe">
  <xs:attribute name = "zeilennummer" type = "xs:normalizedString"/>
  <xs:attribute name = "html-name" type = "xs:normalizedString"/>
</xs:attributeGroup>

<xs:attributeGroup name = "NameTextKlasseHTMLAttrGruppe">
  <xs:attribute name = "name" use = "required" type = "xs:normalizedString"/>
  <xs:attribute name = "text" type = "xs:normalizedString"/>
  <xs:attribute name = "klasse" type = "xs:normalizedString"/>
  <xs:attribute name = "html-name" type = "xs:normalizedString"/>
</xs:attributeGroup>

<!-- 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 = "AbsenderBerichtspflichtigerTyp"/>
<xs:element name = "empfaenger" type = "EmpfaengerBerichtsempfaengerTyp"/>
<xs:element name = "nachricht" type = "NachrichtTyp"/>
<xs:element name = "erhebung" type = "ErhebungTyp"/>
<xs:element name = "berichtszeitraum" type = "BerichtszeitraumTyp"/>

```

```
<xs:element name = "berichtspflichtiger" type = "AbsenderBerichtspflichtigerTyp"/>
<xs:element name = "berichtsempfaenger" type = "EmpfaengerBerichtsempfaengerTyp"/>
<xs:element name = "datensegment" type = "DatensegmentTyp"/>
<xs:element name = "segment" type = "SegmentTyp"/>

<!-- Root Element
-->

<xs:element name = "DatML-RAW-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 = "nachricht" maxOccurs = "unbounded"/>
      <xs:element name = "anzahl" type = "xs:positiveInteger" minOccurs = "0"/>
    </xs:sequence>
    <xs:attribute name = "version" use = "required">
      <xs:simpleType>
        <xs:restriction base = "xs:token">
          <xs:enumeration value = "2.0"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:complexType>
</xs:element>
</xs:schema>
```