

Data Markup Language/RAW (DatML/RAW) 1.0

Spezifikation

Autor: Michael Schäfer, michael.schaefer@destatis.de, +49(0)611/75-3652

Stand: 11.09.2002

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.
- DatML/MAP für das Mappen von Rohdaten aus DatML/RAW, aus HTML-Formularen (POST) und aus sequentiellen Dateien auf DatML/SET, DB-Schemata und sequentielle Dateien, und (teilweise) umgekehrt.
- 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, Mapping 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 1.0.

1. Inhaltsverzeichnis

1. Inhaltsverzeichnis	2
2. Einführung	5
2.1. Verwendung	5
2.1.1. Ein generischer Dokumenttyp	5
2.1.2. Anwendungen von DatML/RAW	5
2.1.2.1. Spezifikation einer Anwendung	5
2.1.2.2. Softwareanforderungen, Schichtenmodell	6
2.1.3. Verfügbarkeit	6
2.2. Struktur	6
2.2.1. Nachrichten	7
2.2.2. Segment, Segmentbaum	7
2.2.3. Datensegment, Segmentpfad	7
2.3. Metadaten	8
2.3.1. Absender und Empfänger	8
2.3.2. Berichtspflichtige und Berichtsempfänger	8
2.3.3. Identifikation	9
2.3.4. Berechtigung	9
2.3.5. Adreßangaben	9
2.3.6. Kontaktinformationen	9
2.3.7. Korrekturen	9
2.3.8. Statistischer Kontext	9
2.3.9. Verarbeitungsoptionen	10
2.3.9.1. Testunterstützung	10
2.3.9.2. Empfangsbestätigung	10
2.3.10. Protokollinformationen	11
2.4. Daten (Statistische Informationen)	11
2.4.1. Datensätze	11
2.4.2. Merkmale (Statistische Variablen)	11
2.4.2.1. Einfache Merkmale	11
2.4.2.2. Ordnungsmerkmale	12
2.4.2.3. Hilfsmerkmale	12
2.4.3. Merkmalsgruppen	12
2.4.4. Namen	15
2.4.4.1. Wertebereich	15
2.4.4.2. Namenskonflikte	15
2.4.5. Sortierung	16
2.4.6. Numerische Werte	16
2.4.7. Formatierte Werte	17
2.5. Verarbeitung	18
2.5.1. Verteilung	18
2.5.2. Zerlegung	18
2.6. Namensraum	19
2.7. Weiteres	20
2.7.1. Informationen aus Online-Erhebungen	20
2.7.2. Unterstützung zeilenbasierter Formate	20
2.7.2.1. Zeilenbasierte Ablage	20
2.7.2.2. Mappen von Typen auf Zeilen	20
2.7.3. Vorgeschriebene Elemente	21
2.7.4. Einfügen von Daten	22

3. Elementtypen	23
3.1. Dokument (Top-Level)	23
3.1.1. Element <DatML-RAW-D>	23
3.1.2. Element <nachricht>	23
3.1.3. Element <segment>	24
3.2. Absender und Empfänger eines Dokuments	26
3.2.1. Element <absender>	26
3.2.2. Element <empfaenger>	26
3.3. Identifikation und Kontaktangaben	26
3.3.1. Element <adresse>	26
3.3.2. Element <anrede>	26
3.3.3. Element <bundesland>	27
3.3.4. Element <email>	27
3.3.5. Element <fax>	27
3.3.6. Element <identifikation>	27
3.3.7. Element <identitaet>	27
3.3.8. Element <kontakt>	27
3.3.9. Element <korrektur>	28
3.3.10. Element <kreis>	28
3.3.11. Element <land>	28
3.3.12. Element <nachname>	28
3.3.13. Element <niederlassung>	29
3.3.14. Element <organisation>	29
3.3.15. Element <ort>	29
3.3.16. Element <person>	29
3.3.17. Element <postfach>	29
3.3.18. Element <postleitzahl>	30
3.3.19. Element <strasse>	30
3.3.20. Element <titel>	30
3.3.21. Element <telefon>	30
3.3.22. Element <vorname>	30
3.3.23. Element <zeile>	30
3.4. Statistischer Kontext	31
3.4.1. Elemente <beginn>, <ende>	31
3.4.2. Element <berichtsempfaenger>	31
3.4.3. Element <berichtspflichtiger>	31
3.4.4. Element <berichtszeitraum>	32
3.4.5. Element <datenattribute>	32
3.4.6. Element <erhebung>	33
3.4.7. Element <hmm>	33
3.4.8. Elemente <jahr>, <halbjahr>, <quartal>, <monat>, <woche>, <tag>	33
3.4.9. Element <lieferattribute>	34
3.4.10. Element <material>	34
3.5. Daten	34
3.5.1. Element <mm>	34
3.5.2. Element <mmgr>	35
3.5.3. Element <na>	35
3.5.4. Element <omm>	35
3.5.5. Element <satz>	36
3.5.6. Element <wert>	36
3.6. Verarbeitungssteuerung	37

3.6.1. Element <empfangsbestaetigung>	37
3.6.2. Element <optionen>	37
3.6.3. Element <test>	37
3.7. Protokollierung	38
3.7.1. Element <eingang>	38
3.7.2. Element <instanz>	39
3.7.3. Element <protokoll>	39
3.8. Sonstige Elemente	40
3.8.1. Element <anzahl>	40
3.8.2. Element <berechtigung>	40
3.8.3. Element <kennung>	40
3.8.4. Element <memo>	40
3.8.5. Element <name>	40
3.8.6. Element <string>	40
3.8.7. Element <text>	41
3.8.8. Element <version>	41
3.8.9. Element <zusatz>	41
4. Änderungen gegenüber Vorschlag #1	42
5. ToDo 2.0 (Nicht normativ)	43
6. Beispieldokument (Nicht normativ)	44
7. DTD 1.0 (Nicht normativ)	46

2. Einführung

2.1. Verwendung

2.1.1. 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.

2.1.2. Anwendungen von DatML/RAW

Eine 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 jede Anwendung von DatML/RAW ein Subset des Dokumenttyps in Form einer maßgeschneiderten DTD oder eines entsprechenden anderen Schemas zu definieren. Eine Parametrisierung der DatML/RAW-DTD ist für die Zukunft geplant. Unabhängig von technischen Aspekten muß jede DatML/RAW-Anwendung sicherstellen, daß ein Subset vollkommen konform mit dem vollständigen Dokumenttyp ist und alle fehlerfreien Dokumente dieser Anwendung gegen beide Varianten fehlerfrei validierbar sind.

2.1.2.1. 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 bzw. des Empfangs für Dritte (sogenannte Mandantenfähigkeit);
 - Unterstützung von Verarbeitungsoptionen (s. 2.3.9);
 - 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).

2.1.2.2. Softwareanforderungen, Schichtenmodell

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. 2.5.1).
- Aufgabe der **Anwendungsschicht** wäre die erhebungsspezifische Verarbeitung der Einzellieferungen, z.B. die Abbildung auf andere Dateiformate und die Plausibilisierung, basierend auf externen Beschreibungsobjekten (z.B. DatML/SSP, DatML/MAP).
- 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 E-Mail-Versand oder in Form der Vorbereitung und Veranlassung manueller Vorgänge (Ausdruck, Postversand).

2.1.3. Verfügbarkeit

DatML/RAW steht in einer deutschen und einer englischen Variante zur Verfügung, die sich durch die unterschiedliche Benennung der Elemente, Attribute und vordefinierten Attributwerte unterscheiden.

2.2. Struktur

Jedes DatML/RAW-Dokument muß folgende Komponenten enthalten: den Absender (**<absender>**), den Empfänger (**<empfaenger>**) und *mindestens eine* Nachricht (**<nachricht>**, s.2.2.1). Jede Nachricht kann in Teilnachrichten, sogenannte *Segmente*, unterteilt sein (**<segment>**, s.2.2.2).

Die Unterteilung eines Dokumentes in Nachrichten und einer Nachricht in Segmente ist ein flexibler Mechanismus der 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.

2.2.1. Nachrichten

Nachrichten enthalten statistische Rohdaten und 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 Metadaten redundanzfrei abzulegen.

2.2.2. Segment, Segmentbaum

Ein Segment kann entweder weitere Segmente oder Datensätze enthalten. Durch die hierarchische Anordnung von Segmentknoten entsteht ein sogenannter *Segmentbaum*. Die logische Wurzel des Segmentbaumes ist der Nachrichtenknoten (**<nachricht>**), der die gleichen Eigenschaften wie ein Segmentknoten hat, mit folgender Ausnahme: Jeder Segmentknoten enthält mindestens ein Metadatum, im Nachrichtenknoten hingegen sind Metadaten optional. Der Nachrichtenknoten kann daher logisch wie ein Segmentknoten behandelt werden. Die Blätter des Segmentbaumes sind die Segmentknoten (oder in unsegmentierten Nachrichten der Nachrichtenknoten), die keine weiteren Segmentknoten mehr enthalten.

2.2.3. Datensegment, Segmentpfad

Ein Blatt eines Segmentbaumes enthält wiederum mindestens ein Metadatum und – anstelle weiterer Segmentknoten – ein sogenanntes *Datensegment*. Ein Datensegment ist der logische Abschnitt eines Segmentes, der die Datensätze enthält, und besteht entweder aus einer Folge von Datensätzen (**<satz>**) oder aus einem Ordnungsmerkmal (**<omm>**), dem die Datensätze nachgeordnet sind.

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. 2.3.8). Eine semantische Überlappung ist nicht erlaubt, d.h. jedes Metadatum darf innerhalb einer Nachricht in einem einzelnen Segmentpfad nur ein einziges Mal vorkommen.

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. 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 Informationen werden als Metadaten bezeichnet und dadurch von den Daten bzw. statistischen Informationen unterschieden (s. 2.4). Metadaten beschreiben einerseits die gelieferten statistischen Informationen und erhöhen deren Qualität und die ihrer Verarbeitung, andererseits finden sie Verwendung in Neben- und Unterstützungsprozessen wie Identifikation, Adreßverwaltung und Plausibilisierung und sind für die Automatisierung statistischer Erhebungs- und Aufbereitungsprozesse unerlässlich.

2.3.1. 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.

2.3.2. Berichtspflichtige und Berichtsempfänger

Es können Daten beliebig vieler Berichtspflichtiger an beliebig viele Berichtsempfänger (die erhebenden Stellen) in einem Dokument übermittelt werden. Per Default sind der Berichtspflichtige mit dem Absender und der Berichtsempfänger mit dem Empfänger des Dokumentes identisch. Berichtspflichtiger und –empfänger müssen daher nur angegeben werden, wenn sie nicht Absender bzw. Empfänger des Dokumentes sind. (s. 2.3.8). Für Änderungen der Adreß- und Kontaktinformationen eines Berichtspflichtigen steht der Knoten **<korrektur>** zur Verfügung.

2.3.3. Identifikation

Die Identifikation der an einem Lieferverfahren Beteiligten – s. 2.3.1 und 2.3.2 – 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. 2.3.4), 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.

2.3.4. 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.

2.3.5. 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 Muß-Angaben (PLZ, Ort) als leere Elemente abgelegt werden, wenn keine Daten für sie vorhanden sind, aber ein valides Dokument erzeugt werden soll.

2.3.6. Kontaktinformationen

Kontaktinformationen beschreiben den Zugang zu Ansprechpartnern bei den Verfahrensbeteiligten. Analog zur Identifikation stehen wahlweise ein typbasiertes (**<identitaet>**, **<telefon>**, **<email>**, **<fax>**) und ein zeilenbasiertes Modell (**<zeile>**) zur Verfügung.

2.3.7. Korrekturen

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

2.3.8. 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.

2.2.3). 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:

<i>Metadatum</i>	<i>Elementtyp</i>	<i>Angabe</i>	<i>Default</i>
Erhebung	<erhebung>	muß	-
Berichtszeitraum	<berichtszeitraum>	muß	-
Berichtspflichtiger	<berichtszeitraum>	optional	Absender
Berichtsempfänger	<berichtszeitraum>	optional	Empfänger
Materialbezeichnung	<material>	optional	-
Lieferattribute	<lieferattribute>	optional	-
Datenattribute	<datenattribute>	optional	-
Hilfsmerkmal	<hmm>	optional	-

2.3.9. 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>**.

2.3.9.1. Testunterstützung

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

2.3.9.2. Empfangsbestätigung

Das 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 Defaultwert für Versandart und Format ist jeweils **"default"**, d.h., bei Abwesenheit expliziter Werte werden diese von der Anwendung gesetzt.

HINWEIS: Die Nutzbarkeit von Verarbeitungsoptionen setzt voraus, daß sie empfängerseitig in die Verarbeitungssoftware implementiert sind (s. 2.1.2.2).

2.3.10. Protokollinformationen

Protokollinformationen (**<protokoll>**) dienen dazu, Lieferwege, Verarbeitungsstufen und historische Daten eines DatML/RAW-Dokumentes zu dokumentieren. Dateneingang (**<eingang>**) und Dokumentinstanzen (**<instanz>**) werden separat protokolliert. Während für Dokumentinstanzen nur eine historische Protokollierung möglich ist, können für den Dateneingang auch Details wie Lieferweg, Lieferdatenträger und Lieferformat festgehalten werden, wodurch z.B. Online-Meldungen per HTML-Formular, Dateiübertragungen und Datenträgerlieferungen unterschieden werden können.

Protokollinformationen sind optional; Zulassung, Erzeugung und Auswertung liegen im Regelungsbereich der DatML/RAW-Anwendung.

2.4. 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.

2.4.1. Datensätze

Ein statistischer Datensatz stellt meist eine Informationseinheit dar, die zusammenhängende Daten über ein Erhebungsobjekt enthält. In diesem Sinne können Datensätze auch innerhalb DatML/RAW verwendet werden. Die allgemeinere Funktion eines DatML/RAW-Datensatzes (**<satz>**) ist die eines Gültigkeitsbereiches für statistische Merkmale; sie müssen innerhalb eines Datensatzes eindeutig sein (s. 2.4.2).

2.4.2. 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.

2.4.2.1. 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. 2.4.3) 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.

2.4.2.2. 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 Bestandteil nicht der Datensätze, sondern einer beliebigen Anzahl von Datensätzen übergeordnet. Eine Folge von Datensätzen wird jeweils genau einer Ausprägung – einem Wert – eines Ordnungsmerkmals (und ggf. denen übergeordneter Ordnungsmerkmal) 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 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) und Ordnungsmerkmale der gleichen Hierarchiestufe 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.

2.4.2.3. Hilfsmerkmale

Hilfsmerkmale (**<hmm>**) zählen nicht zu den Daten wie einfache und Ordnungsmerkmale, sondern erlauben es, anwendungsabhängige Metadaten in den statistischen Kontext zu integrieren. Sie sind daher auch nicht in Datensegmenten zulässig, sondern werden wie die Metadaten des statistischen Kontextes in einem Nachrichten- oder Segmentknoten gespeichert. Wie Ordnungsmerkmale haben sie einen Namen (Attribut **name**) und einen Wert (**<wert>**), der nicht leer sein darf.

2.4.3. 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. Eine 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.

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> 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.

Dabei muß beachtet werden:

- 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.
- In einem *Datensegment* müssen alle Merkmalsgruppen mit demselben Merkmalsnamen auch denselben Wert für das Attribut **index** aufweisen, d.h. dieselbe Indizierungsmethode benutzen.
- In einem Datensatz muß für jede Instanz einer Merkmalsgruppe ein eindeutiger Indexwert ermittelt werden können.

- Die Instanzen einer Merkmalsgruppe müssen weder nach dem Index sortiert noch in ununterbrochener Folge in einem Datensatz stehen.

Eine DatML/RAW-Anwendung muß darüberhinaus 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">03</mm>
  <mm name="insgesamt">...</mm>
  <mm name="belegungstage">...</mm>
  <mm name="patienten">...</mm>
</mmgr>
<mmgr name="intensivbetten" index="2">
  <mm name="abrechnungsmonat">05</mm>
  <mm name="insgesamt">...</mm>
  <mm name="belegungstage">...</mm>
  <mm name="patienten">...</mm>
</mmgr>
```

Indirekter Index; das indexbildende Merkmal hat den Namen „abrechnungsmonat“; die Indizierung weicht in diesem Fall (3,5) vom vorhergehenden Beispiel (1,2) ab:

```
<mmgr name="intensivbetten" index="name(abrechnungsmonat)">
  <mm name="abrechnungsmonat">03</mm>
  <mm name="insgesamt">...</mm>
  <mm name="belegungstage">...</mm>
  <mm name="patienten">...</mm>
</mmgr>
<mmgr name="intensivbetten" index="name(abrechnungsmonat)">
  <mm name="abrechnungsmonat">05</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">maerz</mm> <!-- ergibt 3 -->
  <mm name="insgesamt">...</mm>
  <mm name="belegungstage">...</mm>
  <mm name="patienten">...</mm>
</mmgr>
<mmgr name="intensivbetten" index="name(abrechnungsmonat)">
  <mm name="abrechnungsmonat">mai</mm> <!-- ergibt 5 -->
  <mm name="insgesamt">...</mm>
  <mm name="belegungstage">...</mm>
  <mm name="patienten">...</mm>
</mmgr>

```

2.4.4. Namen

2.4.4.1. 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 internen Namenskonventionen zu genügen.

2.4.4.2. Namenskonflikte

Namensgleichheit zwischen Merkmalen liegt vor, wenn die Attribute **name** und **klasse** jeweils gleiche Werte haben, und muß in den folgenden Fällen einen Namenskonflikt auslösen:

- *Innerhalb eines Segmentpfades* zwischen zwei Hilfsmerkmalen;
- *Innerhalb eines Datensegmentes* zwischen einem Ordnungsmerkmal und einem einfachen Merkmal;
- *Innerhalb eines Datensatzes* zwischen zwei einfachen Merkmalen, wenn mindestens eines davon nicht Bestandteil einer Merkmalsgruppe ist;
- *Innerhalb einer Merkmalsgruppe* zwischen zwei einfachen Merkmalen.

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 invalidiert.

2.4.5. 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. 2.4.2.2).

2.4.6. 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:

- Das allgemeine Format ist `[w][s]a[e[s]n][w]` (Angaben in `[]` sind optional):
 - `w` : Whitespace gemäß XML-Spezifikation, also Leerzeichen (X'20'), Tabulatorzeichen (X'09'), Wagenrücklauf (X'0D') und Zeilenvorschub (X'0A')
 - `s` : Vorzeichen („+“, „-“)
 - `a` : Betrag (0-9, optional Tausender-Trennzeichen und Dezimalzeichen)
 - `e` : Exponentenkennzeichen („e“, „E“)
 - `n` : Exponent (eine natürliche Zahl)
- Leerzeichen sind nur am Anfang und am Ende erlaubt und werden dort ignoriert.
- Insignifikante (führende und folgende) Nullen sind erlaubt und werden ignoriert.
- Insignifikanter (führender und folgender) Whitespace ist erlaubt und wird ignoriert.
- Dezimalzeichen lt. **<datenattribute dezimalzeichen=“...“/>**.
- Tausender-Trennzeichen lt. **<datenattribute tausender-trennzeichen=“...“/>**.
- Dezimalzeichen und Tausender-Trennzeichen müssen verschieden sein.
- Das erste von Whitespace 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. den Dezimalzeichen (Null wird als Wert vor dem Dezimalzeichen angenommen) handeln. Einem Tausender-Trennzeichen müssen drei Ziffern folgen. Nach dem Dezimalzeichen muß eine Ziffer folgen, danach dürfen bis zum Ende des Betrages nur noch Ziffern folgen. Folgen keine Ziffern mehr, wird das Dezimalzeichen ignoriert (keine Nachkommastellen). Der Betrag endet mit dem Exponentenkennzeichen oder Whitespace; endet er mit Whitespace, müssen alle evtl. folgenden Zeichen ebenfalls Whitespace sein.
- Nach dem Exponentenkennzeichen darf nur ein optionales Vorzeichen gefolgt vom Exponentenbetrag gefolgt von optionalem Whitespace 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.

2.4.7. Formatierte Werte

Mit dem Element **<string>** ist es möglich, formatierte Zeichenketten zu verwenden. Die erforderlichen Attribute **format** und **klasse** liefern ein Formatschema und den Namen der Klasse, zu der das Formatschema gehört. Aufgrund des Klassennamens kann das Formatschema interpretiert werden. Die Tabelle zeigt die möglichen Attributwerte für Klassen und Formatschemata:

klasse	format
datum	<p>Bedeutung: Ein Einzeldatum oder eine Zeitspanne aus zwei Einzeldaten. S. 3.4.3.</p> <p>Formatschema: <i>datum</i>[-<i>datum</i>] <i>datum</i> : [j4]j2 [h q][monat[<i>tag</i>[<i>std</i>[<i>min</i>[<i>sek</i>]]]]]</p> <p><i>j4</i> : jjjj, Jahr vierstellig <i>j2</i> : jj, Jahr zweistellig <i>h</i> : h, Halbjahr i.d.F. hn Hn, n=1..2 <i>q</i> : q, Quartal i.d.F. qn Qn, n=1, n=1..4 <i>monat</i> : mm, Monat zweistellig <i>tag</i> : tt, Tag zweistellig <i>std</i> : hh, Stunde zweistellig <i>min</i> : mm, Minute zweistellig <i>sek</i> : mm, Sekunde zweistellig</p> <p>Beispiele: jjjjmm : Jahr vierstellig und Monat, 200205 jj : Jahr zweistellig, 02 jj-jj : Jahresspanne zweistellig, 01-02 jjq : Jahre zweistellig + Quartal, 02q2, 02Q2 jjjjh : Jahre vierstellig + Halbjahr, 2002h1, 2002H1</p>

2.5. Verarbeitung

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

2.5.1. 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 entsprechenden Stufen des Verarbeitungsprozesses statt (s. 2.1.2.2).

- *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. 2.3.8), 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.

2.5.2. Zerlegung

Die ursprüngliche Segmentierung eines Dokuments muß bei einer Zerlegung nicht notwendigerweise erhalten bleiben. Es ist z.B. überlegenswert, 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 Metadatensatz.
- 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, beginnend mit einem Ordnungsmerkmal oder einem Satz, liegt im letzten Segmentknoten des Segmentpfades, 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.

2.6. Namensraum

Für DatML/RAW wurden entsprechend den durch das [Worldwide Web Consortium \(W3C\)](#) veröffentlichten Empfehlungen für [Namensräume \(Namespaces\)](#) folgende Namensräume definiert:

Für die deutsche Variante: <http://www.destatis.de/schema/DatML-RAW/1.0/D>.

Für die englische Variante: <http://www.destatis.de/schema/DatML-RAW/1.0/E>.

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 **<identitaet>** erlaubten Elementtypen mit Dateninhalt (PCDATA)
- Alle in **<kontakt>** erlaubten Elementtypen mit Dateninhalt (PCDATA)
- Alle Merkmalstypen: **<mm>**, **<hmm>**, **<omm>**

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 das *Mappen typbasierter auf zeilenbasierte Einträge*.

HINWEIS: Es wird empfohlen, typbasierte Datenmodelle und die entsprechenden DatML/RAW-Strukturen zu benutzen. Zeilenbasierte Datenmodelle werden nur aus dem Grund unterstützt, DatML/RAW in existierenden Erhebungen leichter implementieren zu können.

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.

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

2.7.2.2. Mappen von Typen auf Zeilen

Das explizite Mappen 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 zu mappen, 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 gemappt.
- Wenn mehr als ein Elementtyp auf die selbe Zeile gemappt wird, werden die Elementinhalte – in der Reihenfolge ihres Auftretens oder in einer definierten Reihenfolge – zu einem Resultatstring konkateniert; falls notwendig, sollte ein zusätzlich eingefügtes Leerzeichen zwei Werte trennen.
- Für Zwecke des Abgleiches werden aus dem Resultatstring 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 das Mapping anhand der vorhandenen Werte spezifiziert und ausgeführt werden. Die Korrekturen selbst sollten typbasiert sein und keine Mappinginformationen verwenden. Es können so keine Werte korrigiert werden, die nicht als vorhanden geliefert wurden, da für sie kein Mapping existiert. Falls das Mapping sowohl für die vorhandenen Werte als auch für die Korrekturen spezifiziert ist, muß es 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 validen 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.

Für dieses Vorgehen gibt es vor allem technische Gründe. Die Sprachmittel einer XML-DTD erlauben nur die Verwendung weniger und zudem kontextinsensitiver Restriktionen. Eine Redefinition von DatML mit mächtigeren Schemasprachen, wie XML Schema oder RELAX, ist geplant.

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, doch müssen sie immer eindeutig unterschieden werden können. Daten dürfen deshalb keine Zeichen enthalten, die das korrekte Erkennen des Markups 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 Zeichenlitterale. 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 Dokument \(Top-Level\)](#)

[3.2 Absender und Empfänger eines Dokuments](#)

[3.3 Identifikation und Kontaktangaben](#)

[3.4 Statistischer Kontext](#)

[3.5 Daten](#)

[3.6 Verarbeitungssteuerung](#)

[3.7 Protokollierung](#)

[3.8 Sonstige Elemente.](#)

3.1. Dokument (Top-Level)

3.1.1. Element <DatML-RAW-D>

Klasse: Root-Element

Inhalt: *Elemente*

Modell: *Sequenz.*

memo	0-1
optionen	0-1
protokoll	0-1
absender	1
empfaenger	1
nachricht	1-n
anzahl	0-1

Attrib: **version**

erforderlich char_const 1.0

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.

3.1.2. Element <nachricht>

Inhalt: *Elemente*

Kinder *Sequenz.*

memo	0-1
<i>Auswahl:</i>	0-n
erhebung	1

berichtszeitraum	1
berichtspflichtiger	1
berichtsempfaenger1	
material	1
lieferattribute	1
datenattribute	1
hmm	1
<i>Auswahl:</i>	
omm	1
satz	1-n
segment	1-n

Attrib: -

Eine Nachricht besteht aus einem optionalen Kommentar und einer Auswahl optionaler Metadaten, gefolgt von entweder einem Ordnungsmerkmal (**<omm>**) oder einer Folge von Datensätzen (**<satz>**) oder einer Folge von Segmenten (**<segment>**).

Die Reihenfolge der Metadaten ist beliebig, jedoch darf jedes Metadatum mit Ausnahme von **<hmm>** in einem Segmentpfad nur einmal vorkommen, egal, ob es sich in einem Nachrichten- oder einem Segmentknoten befindet; s. 2.3.

3.1.3. Element **<segment>**

Inhalt: *Elemente*

Kinder *Sequenz:*

memo	0-1
<i>Auswahl:</i>	
erhebung	1
berichtszeitraum	1
berichtspflichtiger	1
berichtsempfaenger1	
material	1
lieferattribute	1
datenattribute	1
hmm	1
<i>Auswahl:</i>	
omm	1
satz	1-n
segment	1-n

Attrib: -

Ein Segment besteht aus einem optionalen Kommentar und *mindestens einem* Metadatum, gefolgt von entweder einem Ordnungsmerkmal (**<omm>**) oder einer Folge von Datensätzen (**<satz>**) 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 mit Ausnahme von **<hmm>** auf

einem Segmentpfad nur einmal vorkommen, egal, ob es sich in einem Nachrichten- oder einem Segmentknoten befindet; s. 2.2.3.

Segmente können hierarchisch angeordnet sein, da das Element **<segment>** eine Rekursion auf sich selbst erlaubt. Ein Satzknoten (**<satz>**) oder ein Ordnungsmerkmal (**<omm>**) 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 die Daten enthält.

Beispiele:

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

<segment>
  <erhebung>
    <kennung klasse="evas">11111</kennung>
  </erhebung>
  <segment>
    <berichtszeitraum>
      <jahr>2001</jahr>
      <monat>01</monat>
    </berichtszeitraum>
    <satz> ... </satz>
  </segment>
  <segment>
    <berichtszeitraum>
      <jahr>2001</jahr>
      <monat>02</monat>
    </berichtszeitraum>
    <satz> ... </satz>
  </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">11111</text>
      <text>Einfuhr</text>
    </erhebung>
    <satz> ... </satz>
  </segment>
  <segment>
    <erhebung>
      <kennung klasse="EVAS">11112</text>
      <text>Ausfuhr</text>
    </erhebung>
    <satz> ... </satz>
  </segment>
</nachricht>
```

3.2. Absender und Empfänger eines Dokuments

3.2.1. Element <absender>

Inhalt: *Elemente*

Modell: *Sequenz:*

kennung	1
berechtigung	0-1
identifikation	1
kontakt	0-1
korrektur	0-1
memo	0-1

Attrib: -

3.2.2. Element <empfaenger>

Inhalt: *Elemente*

Modell: *Sequenz:*

kennung	0-1
identifikation	1
kontakt	0-1
memo	0-1

Attrib: -

3.3. Identifikation und Kontaktangaben

3.3.1. Element <adresse>

Inhalt: *Elemente*

Modell: *Sequenz:*

strasse	0-1
postfach	0-1
postleitzahl	1
ort	1
kreis	0-1
bundesland	0-1
land	0-1
zusatz	0-1

Attrib: -

3.3.2. Element <anrede>

Inhalt: *Daten*

Attrib: **html-name**

optional char

zeilennummer
optional int

3.3.3. Element <bundesland>

Inhalt: *Daten*
Attrib: **html-name**
optional char
zeilennummer
optional int

3.3.4. Element <email>

Inhalt: *Daten*
Attrib: **html-name**
optional char
zeilennummer
optional int

3.3.5. Element <fax>

Inhalt: *Daten*
Attrib: **html-name**
optional char
zeilennummer
optional int

3.3.6. Element <identifikation>

Inhalt: *Elemente*
Modell: *Auswahl:*
 zeile 1-n
 Sequenz:
 identitaet 1
 adresse 0-1

Attrib: -

3.3.7. Element <identitaet>

Inhalt: *Elemente*
Modell: *Sequenz:*
 organisation 0-1
 person 0-1

Attrib: -

3.3.8. Element <kontakt>

Inhalt: *Elemente*

Modell: *Auswahl:*

zeile	1-n
<i>Sequenz:</i>	
identitaet	0-1
<i>Auswahl:</i>	
telefon	1
email	1
fax	1

Attrib: -

Kontaktadresse für Rückfragen, zuständige Person, etc.

3.3.9. Element <korrektur>

Inhalt: *Elemente*

Modell: *Sequenz:*

identifikation	0-1
kontakt	0-1

Attrib: -

Das Element <korrektur> erlaubt es, Änderungen der Adreß- und Kontaktinformationen und der Identität (z.B. Änderung des Firmennames) eines Absenders oder Berichtspflichtigen zu übermitteln.

3.3.10. Element <kreis>

Inhalt: *Daten*

Attrib: **html-name**

optional char

zeilennummer

optional int

<kreis> bezeichnet die Verwaltungseinheit Kreis; in anderen Ländern könnte dies z.B. einem Bezirk, Distrikt oder Departement entsprechen.

3.3.11. Element <land>

Inhalt: *Daten*

Attrib: **html-name**

optional char

zeilennummer

optional int

<land> bezeichnet einen Nationalstaat.

3.3.12. Element <nachname>

Inhalt: *Daten*

Attrib: **html-name**

optional char
zeilennummer
optional int

3.3.13. Element <niederlassung>

Inhalt: *Daten*

Attrib: **html-name**
optional char
zeilennummer
optional int

3.3.14. Element <organisation>

Inhalt: *Elemente*

Modell: *Sequenz:*

name	1
niederlassung	0-1
zusatz	0-1

Attrib: -

Eine Firma, Behörde, Institution, Einrichtung oder andere Form von juristischer Person (im Ggs. zu einer natürlichen Person).

3.3.15. Element <ort>

Inhalt: *Daten*

Attrib: **html-name**
optional char
zeilennummer
optional int

Eine Stadt, Gemeinde etc.

3.3.16. Element <person>

Inhalt: *Elemente*

Modell: *Sequenz:*

anrede	0-1
titel	0-1
vorname	0-1
nachname	1
zusatz	0-1

Attrib: -

3.3.17. Element <postfach>

Inhalt: *Daten*

Attrib: **html-name**

optional *char*
zeilennummer
optional *int*

3.3.18. Element <postleitzahl>

Inhalt: *Daten*
Attrib: **html-name**
optional *char*
zeilennummer
optional *int*

3.3.19. Element <strasse>

Inhalt: *Daten*
Attrib: **html-name**
optional *char*
zeilennummer
optional *int*

3.3.20. Element <titel>

Inhalt: *Daten*
Attrib: **html-name**
optional *char*
zeilennummer
optional *int*

3.3.21. Element <telefon>

Inhalt: *Daten*
Attrib: **html-name**
optional *char*
zeilennummer
optional *int*

3.3.22. Element <vorname>

Inhalt: *Daten*
Attrib: **html-name**
optional *char*
zeilennummer
optional *int*

3.3.23. Element <zeile>

Inhalt: *Daten*
Attrib: **html-name**
optional *char*
zeilennummer
optional *int*

3.4. Statistischer Kontext

Die nachfolgenden Elemente beschreiben den statistischen Kontext von Rohdaten, wie z.B. Erhebung und Berichtszeitraum.

3.4.1. Elemente <beginn>, <ende>

Inhalt: *Elemente*

Modell: *Sequenz:*

jahr	1	
<i>Auswahl:</i>		
halbjahr	0-1	
quartal	0-1	
<i>Sequenz:</i>		
monat		1
tag		0-1
woche	0-1	

<beginn> gibt den Beginn, <ende> das einschließliche Ende eines Berichtszeitraumes an.

3.4.2. Element <berichtsempfaenger>

Inhalt: *Elemente*

Modell: *Sequenz:*

kennung	0-1
identifikation	1
kontakt	0-1
memo	0-1

Attrib: -

Berichtsempfänger ist i.d.R. ein Statistisches Landesamt oder das Statistische Bundesamt. Per Default ist der Berichtsempfänger mit dem Empfänger des Dokumentes identisch.

3.4.3. Element <berichtspflichtiger>

Inhalt: *Elemente*

Modell: *Sequenz:*

kennung	1
berechtigung	0-1
identifikation	1
kontakt	0-1
korrektur	0-1
memo	0-1

Attrib: -

Per Default ist der Berichtspflichtige mit dem Absender identisch.

3.4.4. Element <berichtszeitraum>

Inhalt: *Elemente*

Modell: *Auswahl:*

string	1	
<i>Auswahl:</i>		
<i>Sequenz:</i>		
jahr	1	
<i>Auswahl:</i>		
halbjahr		0-1
quartal		0-1
<i>Sequenz:</i>		
monat		1
tag		0-1
woche		0-1
beginn	1	
ende	1	

<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 i.V.m. 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. 3.8.4, Element **<string>**.

Die Festlegung zulässiger Datumsangaben und weitere Regeln (invertierte Spannen,..) sind Sache der Anwendung.

3.4.5. Element <datenattribute>

Inhalt: *leer*

Attrib: **dezimalzeichen**

optional char ‘
tausender-trennzeichen
optional char ‘

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.

3.4.6. Element **<erhebung>**

Inhalt: *Elemente*

Modell: *Sequenz:*

kennung	1
text	0-1
berechtigung	0-1

Attrib: -

Eine Erhebungsangabe hat eine ID (**<kennung>**) und einen optionalen Namen (**<name>**). Stammt die ID 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.

3.4.7. Element **<hmm>**

Inhalt: *Elemente*

Modell: *Sequenz:*

memo	0-1
wert	1

Attrib: **name**

erforderlich char

html-name

optional char

text

optional char

klasse

optional char

Der Elementtyp **<hmm>** beschreibt ein statistisches Hilfsmerkmal (s. 2.4.2.3) und ist unter denen des statistischen Kontextes der Einzige mit beliebigem Vorkommen. Das Attribut **name** identifiziert das Merkmal und muß immer angegeben werden.

3.4.8. Elemente **<jahr>**, **<halbjahr>**, **<quartal>**, **<monat>**, **<woche>**, **<tag>**

Inhalt: *Daten*

Attrib: -

Die Elemente bestimmen Bestandteile von Datumsangaben. Der Inhalt ist eine numerischer Wert; **<halbjahr>**, **<quartal>**, **<monat>** und **<woche>** werden in Relation zum Jahr, **<tag>** in Relation zum Monat angegeben, **<jahr>** enthält immer einen vierstelligen Wert.

3.4.9. Element **<lieferattribute>**

Inhalt: *leer*

Attrib: **lieferstatus**

optional *enum* default | erstlieferung | nachlieferung | korrektur

liefernummer

optional *int*

<lieferattribute> beschreibt den Status einer Datenlieferung, d.h. Erst-, Korrektur oder Nachlieferung (Attribut **lieferstatus**) und eine damit ggf. verbundene Liefernummer (Attribut **liefernummer**). Verwendung und Konsistenzprüfung ist Sache der Anwendung.

3.4.10. Element **<material>**

Inhalt: *leer*

Modell: *Sequenz:*

name 0-1

kennung 0-1

version 0-1

Attrib: -

Bezeichnung und Version des Liefermaterials. Mit dem Attribut **klasse** kann eine für die Bezeichnung verwendete Klassifikation angegeben werden.

3.5. Daten

3.5.1. Element **<mm>**

Inhalt: *Elemente*

Modell: *Sequenz:*

memo 0-1

Auswahl:

wert 1

na 1

Attrib: **name**

erforderlich *char*

html-name

optional *char*

text

optional *char*

klasse
optional char

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

3.5.2. Element <mmgr>

Inhalt: *Elemente*

Modell: *Sequenz:*
mm 1-n

Attrib: **name**
erforderlich char
html-name
optional char
text
optional char
klasse
optional char
index
optional char

Einfache Merkmale (<mm>) 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. 2.4.3).

3.5.3. Element <na>

Inhalt: *Daten*

Attrib: -

<na> steht für eine *explizite* „Nichtantwort“; der Elementinhalt ist beliebig und erlaubt daher z.B. die Angabe eines Grundes. Die Auswertung ist anwendungsabhängig. Im Hinblick auf mögliche zukünftige Erweiterungen wird empfohlen, den Inhalt nur als Kommentar zur werten.

3.5.4. Element <omm>

Inhalt: *Elemente*

Modell: *Sequenz:*
memo 0-1
Sequenz: 1-n
wert 1
Auswahl:
omm 1
satz 1-n

Attrib: **name**

	<i>erforderlich</i>	<i>char</i>
html-name		
	<i>optional</i>	<i>char</i>
text		
	<i>optional</i>	<i>char</i>
klasse		
	<i>optional</i>	<i>char</i>

<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 **<segment>** 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>
```

3.5.5. Element **<satz>**

Inhalt: *Elemente*

Modell: *Sequenz.*

mm	1-n
anzahl	0-1

Attrib: **kennung**

optional char

Ein Satz besteht aus mindestens einem Wertmerkmal (**<mm>**) und einer optionalen Anzahlangabe (**<anzahl>**) für Kontrollzwecke.

3.5.6. Element **<wert>**

Inhalt:	<i>Daten</i>		
Attrib:	einheit-basis	<i>optional</i>	<i>enum</i> <u>anzahl</u> k m m2 m3 text
	einheit-text	<i>optional</i>	<i>char</i> Eine Einheitsangabe
	einheit-faktor	<i>optional</i>	<i>int</i> <u>1</u>

Die Einheit eines Wertes kann mit dem Attribut **einheit-basis** angegeben werden. **einheit-basis="text"** bedeutet, daß die Einheitsbezeichnung per Attribut **einheit-text** erfolgt, und ist eine Hintertür für nicht in der Werteliste enthaltene Einheiten. Das Attribut **einheit-faktor** bietet die Möglichkeit, einen Wert als n-faches bzw. n-ten Bruchteil der Basiseinheit anzugeben.

3.6. Verarbeitungssteuerung

3.6.1. Element <empfangsbestaetigung>

Inhalt:	<i>leer</i>		
Attrib:	versandart	<i>optional</i>	<i>enum</i> <u>default</u> email post
	format	<i>optional</i>	<i>enum</i> <u>default</u> xml text
	betreff	<i>optional</i>	<i>char</i>

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.

3.6.2. Element <optionen>

Inhalt:	<i>Elemente</i>		
Modell:	<i>Sequenz.</i>		
	test		0-1
	empfangsbestaetigung		0-1

Attrib: -

Einstellungen und Benutzeroptionen.

3.6.3. Element <test>

Inhalt:	<i>leer</i>		
Attrib:	kennung	<i>optional</i>	<i>char</i>

<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.

3.7. Protokollierung

3.7.1. Element <eingang>

Inhalt: *leer*

Attrib: **datum**

optional *char* Datum i.d.F. jjjjmmtt

uhrzeit

optional *char* Uhrzeit i.d.F. hhmmss oder hhmm

anwendung

optional *char* Anwendung, die die Lieferdaten erzeugte.

liefermedium

optional *enum* undefiniert|internet|datentraeger

datentraegertyp

optional *enum* undefiniert|diskette|cdrom|dvd|kassette|magnetband|fragebogen

datentraeger-id

optional *char* z.B. Volume-Nr.

lieferformattyp

optional *enum* undefiniert|datensatz|csv|xml|html

lieferformatname

optional *char* Dokumenttyp bei XML

fragebogen

optional *char* Name des Fragebogens

dateiname

optional *char* Name der Eingangsdatei

<eingang> ermöglicht die Dokumentierung des Zeitpunktes, des Weges und weiterer Details des Dateneingangs. Erzeugung der Dokumentinstanz und 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.

Das Attribut **liefermedium** gibt an, ob die Lieferdaten online per Internet oder auf einem physikalischen Datenträger geliefert wurden. Für den zweiten Fall erlauben die Attribute **datentraegertyp** und **datentraeger-id**, den Typ und die *individuelle* Identifikation des Datenträgers anzugeben.

Unabhängig vom Typ des Liefermediums und des Lieferformaten werden – sofern vorhanden – der Dateiname und eine Bezeichnung des Fragebogens in den Attributen **dateiname** bzw. **fragebogen** eingetragen.

lieferformattyp und **lieferformatname** erlauben es, genauere Informationen über das verwendete Format abzulegen.

Beispiel Online-Erhebung (HTML-Formular):

```
<eingang datum="20020506" uhrzeit="195412"
  liefermedium="internet"
  lieferformattyp="html"
  fragenbogen="FB-MZ-02-1"
  >
```

Beispiel Dateilieferung; die Datei `intra-200204` wurde auf Diskette 437161 mit Datensätzen im CSV-Format geliefert und nach DatML/RAW konvertiert:

```
<eingang liefermedium="datentraeger"
  datentraegertyp="diskette"
  datentraeger-id="437161"
  lieferformattyp="csv"
  dateiname="intra-200204"
  >
```

3.7.2. Element <instanz>

Inhalt: *leer*

Attrib: **datum**

optional *char* Datum i.d.F. `jjjjmmtt`

uhrzeit

optional *char* Uhrzeit i.d.F. `hhmmss` oder `hhmm`

anwendung

optional *char* Anwendung, die die Dokumentinstanz erzeugte.

<instanz> erlaubt die Dokumentierung von Datum und Uhrzeit der Erzeugung der Dokumentinstanz sowie der erzeugenden Anwendung.

3.7.3. Element <protokoll>

Inhalt: *Elemente*

Modell: *Sequenz.*

ingang 0-1

instanz 0-n

Attrib: -

<protokoll> erlaubt die Dokumentation des Dateneingangs (<ingang>) und beliebig vieler Dokumentinstanzen (<instanz>).

3.8. Sonstige Elemente

3.8.1. Element <anzahl>

Inhalt: *Daten*

Attrib: -

Ein Kontrollwert für die Anzahlprüfung vorausgegangener Elemente.

3.8.2. Element <berechtigung>

Inhalt: *Daten*

Attrib: **klasse**
optional char default

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.).

3.8.3. Element <kennung>

Inhalt: *Daten*

Attrib: **klasse**
optional char default

3.8.4. Element <memo>

Inhalt: *Daten*

Attrib: -

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

3.8.5. Element <name>

Inhalt: *Daten*

Attrib: **html-name**
optional char
zeilennummer
optional int

Ein beliebiger Name, dessen Bedeutung kontextabhängig ist:

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

3.8.6. Element <string>

Inhalt: *Daten*

Attrib: **format**

erforderlich char
klasse
erforderlich enum undefiniert|datum

<string> ermöglicht die Angabe formatierter Zeichenketten. Z.Z. ist nur die Verwendung von Datumsangaben definiert (s. 2.4.7).

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

3.8.7. Element <text>

Inhalt: *Daten*
Attrib: -

3.8.8. Element <version>

Inhalt: *Daten*
Attrib: -

3.8.9. Element <zusatz>

Inhalt: *Daten*
Attrib: -

4. Änderungen gegenüber Vorschlag #1

1. Das Element **<name>** wurde wieder in die DTD aufgenommen.
2. Das Element **<material>** hat wieder ein element-basiertes Inhaltsmodell mit **<name>**, **<kennung>** und **<version>**.
3. Das Element **<organisation>** wurde durch **<organisation>**, und in dessen Inhaltsmodell **<name>** durch **<name>** ersetzt.
4. Einfügen der Versionsnummer in die Namespace-URI (s. 2.6)

5. ToDo 2.0 (Nicht normativ)

Die folgende Liste nennt – ohne Anspruch auf Vollständigkeit und Relevanz – einige Punkte, die im Hinblick auf eine mögliche Version DatML/RAW 2.0 in der Entwicklungsarbeit Berücksichtigung finden sollten:

- Klassen für Merkmalseigenschaften
- Abbildung von Tabellen
- Überlagerung von Metadaten
- Rekursive Merkmalsgruppen

6. Beispieldokument (Nicht normativ)

Eine Firma liefert zwei Datensätze mit Monatsmeldungen für die Intrahandelsstatistik. Im Dokument werden zwei Ordnungsmerkmale beschrieben: Ursprungsland, mit einem Wert, und diesem nachgeordnet das Zielland, mit zwei Werten. Je Zielland ist ein Datensatz mit vier Merkmalen, 1000 Euro, 1000 Dollar, Kilogramm und Warennummer, abgelegt. Es wird eine Empfangsbestätigung im XML-Format angefordert.

```
<?xml version="1.0" encoding="UTF-8"?>
<DatML-RAW-D version="1.0">
  <optionen>
    <test/>
    <empfangsbestaetigung versandart="email" format="text"/>
  </optionen>
  <absender>
    <kennung>w3s-x88-93z11</kennung>
    <berechtigung>rt56jh7k</berechtigung>
    <identifikation>
      <identitaet>
        <organisation>
          <name>inex Handelskontor</name>
          <niederlassung>Hamburg</niederlassung>
        </firma>
      </identitaet>
      <adresse>
        <strasse>Steinweg 12</strasse>
        <postleitzahl>21201</postleitzahl>
        <ort>Hamburg</ort>
      </adresse>
    </identifikation>
    <kontakt>
      <identitaet>
        <person>
          <anrede>Frau</anrede>
          <vorname>Ingrid</vorname>
          <nachname>Michelsen</nachname>
        </person>
      </identitaet>
      <telefon>030/88 77 5467</telefon>
      <email>michelsen@inex-kontor.de</email>
    </kontakt>
    <korrektur>
      <kontakt>
        <telefon>030/88 77 3427</telefon>
      </kontakt>
    </korrektur>
  </absender>
  <empfaenger>
    <identifikation>
      <identitaet>
        <organisation>
          <name>Statistisches Bundesamt</name>
          <niederlassung>Wiesbaden</niederlassung>
        </firma>
      </identitaet>
    </identifikation>
  </empfaenger>
  <nachricht>
    <segment>
```

```
<erhebung>
  <kennung klasse="evas">1111</kennung>
  <text>Intrahandel</text>
</erhebung>
<berichtszeitraum>
  <jahr>2002</jahr>
  <monat>01</monat>
</berichtszeitraum>
<material>
  <name>INTRA02</name>
  <kennung klasse="EVAS">IN12-01-1</name>
</material>
<om name="ulnd" text="Ursprungsland" klasse="LV2002">
  <wert>001</wert>
  <om name="zlnd" klasse="LV2002" text="Zielland">

    <!-- 1. Wert fuer Zielland -->
    <wert>002</wert>

    <satz>
      <mm name="tdm" text="1000 Euro">
        <wert einheit="euro"
          faktor="1000">568</wert>
      </mm>
      <mm name="tdlr">
        <wert faktor="1000" einheit-text="dollar"
          einheit="text">511</wert>
      </mm>
      <mm name="kilo"
        text="Gewicht der Ware in Kilogramm">
        <wert einheit="k">34768</wert>
      </mm>
      <mm name="wnr" text="Warennummer 9-stellig">
        <wert>123456789</wert>
      </mm>
    </satz>

    <!-- 2. Wert fuer Zielland -->
    <wert>026</wert>

    <satz>
      <mm name="tdm" text="1000 Euro">
        <wert einheit="euro" faktor="1000"
          >568</wert>
      </mm>
      <mm name="tdlr">
        <wert faktor="1000" einheit-text="dollar"
          einheit="text">511</wert>
      </mm>
      <mm name="kilo"
        text="Gewicht der Ware in Kilogramm">
        <wert faktor="1" einheit="k">34768</wert>
      </mm>
      <mm name="wnr" text="Warennummer 9-stellig">
        <wert>123456789</wert>
      </mm>
    </satz>
  </om>
</om>
</segment>
</nachricht>
</DatML-RAW-D>
```

7. DTD 1.0 (Nicht normativ)

Dieser Ausdruck der DTD kann nicht als Referenz verwendet werden.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- *****
Schema      : DatML/RAW
Version    : 1.0
Variante   : D
Edition    : 1
Typ        : XML-DTD
Namespace: http://www.destatis.de/schema/DatML-RAW/D
Stand      : 6. September 2002
Autor      : Michael Schäfer
            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 Berichtszeit-
räume als Erst-, Nach- und Korrekturlieferung übertragen werden.

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

*****
-->

<!-- Root-Element
Das Attribut 'version' muß immer angegeben werden. Absender, Empfänger
und mindestens eine Nachricht sind Mindestbestandteile eines DatML/RAW-
Dokumentes.
-->
<!ELEMENT DatML-RAW-D (memo?, optionen?, protokoll?, absender, empfaenger, nachricht+,
anzahl?)>
<!ATTLIST DatML-RAW-D version (1.0) #REQUIRED
>
<!ELEMENT nachricht (memo?, (erhebung | berichtszeitraum | berichtspflichtiger |
berichtsempfaenger | material | lieferattribute |
datenattribute | hmm)*, (omm | satz+ | segment+))>
<!ELEMENT segment (memo?, (erhebung | berichtszeitraum | berichtspflichtiger |
berichtsempfaenger | material | lieferattribute |
datenattribute | hmm)+, (omm | satz+ | segment+))>

<!-- Absender und Empfänger
-->
<!ELEMENT absender (kennung, berechtigung?, identifikation, kontakt?, korrektur?, memo?)>
<!ELEMENT empfaenger (kennung?, identifikation, kontakt?, memo?)>

<!-- Identifikation und Kontaktangaben
-->
<!ELEMENT adresse (strasse?, postfach?, 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
      >
<!ELEMENT fax (#PCDATA)>
<!ATTLIST fax zeilennummer CDATA #IMPLIED
            html-name CDATA #IMPLIED
          >
<!ELEMENT identifikation (zeile+ | (identitaet, adresse?))>
<!ELEMENT identitaet (organisation?, person?)>
<!ELEMENT kontakt (zeile+ | (identitaet?, (telefon | email | fax)*))>
<!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 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 zeilennummer CDATA #IMPLIED
            html-name CDATA #IMPLIED
          >
<!ELEMENT vorname (#PCDATA)>
<!ATTLIST vorname zeilennummer CDATA #IMPLIED
            html-name CDATA #IMPLIED
          >
<!ELEMENT zeile (#PCDATA)>
<!ATTLIST zeile zeilennummer CDATA #IMPLIED
            html-name CDATA #IMPLIED
          >

<!-- Statistischer Kontext
-->
<!ELEMENT beginn (jahr, (halbjahr | quartal | (monat, tag?) | woche?))>
<!ELEMENT berichtsempfaenger (kennung?, identifikation, kontakt?, memo?)>
<!ELEMENT berichtspflichtiger (kennung, berechtigung?, identifikation, kontakt?,
                               korrektur?, memo?)>
<!ELEMENT berichtszeitraum (string | (jahr, (halbjahr | quartal | (monat, tag?) |
                               woche?) | (beginn, ende)))>
<!ELEMENT datenattribute EMPTY>
<!ATTLIST datenattribute dezimalzeichen CDATA ","
                        tausender-trennzeichen CDATA "."
          >
<!ELEMENT ende (jahr, (halbjahr | quartal | (monat, tag?) | woche?))>
<!ELEMENT erhebung (kennung, text?, berechtigung?)>
<!ELEMENT halbjahr (#PCDATA)>
<!ELEMENT hmm (memo?, wert)>

```

```

<!ATTLIST hmm name CDATA #REQUIRED
            text CDATA #IMPLIED
            klasse CDATA #IMPLIED
            html-name CDATA #IMPLIED
        >
<!ELEMENT jahr (#PCDATA)>
<!ELEMENT lieferattribute EMPTY>
<!ATTLIST lieferattribute lieferstatus (default | erstlieferung | nachlieferung |
            korrektur) "default"
            liefernummer CDATA #IMPLIED
        >
<!ELEMENT material (name?, kennung?, version?)>
<!ELEMENT monat (#PCDATA)>
<!ELEMENT quartal (#PCDATA)>
<!ELEMENT tag (#PCDATA)>
<!ELEMENT woche (#PCDATA)>

<!-- Daten
-->
<!ELEMENT mm (memo?, (wert | na))>
<!ATTLIST mm name CDATA #REQUIRED
            text CDATA #IMPLIED
            klasse CDATA #IMPLIED
            html-name CDATA #IMPLIED
        >
<!ELEMENT mmgr (mm+)>
<!ATTLIST mmgr name CDATA #REQUIRED
            klasse CDATA #IMPLIED
            text CDATA #IMPLIED
            html-name CDATA #IMPLIED
            index CDATA #IMPLIED
        >
<!ELEMENT na (#PCDATA)>
<!ELEMENT omm (memo?, (wert, (omm | satz+))>
<!ATTLIST omm name CDATA #REQUIRED
            text CDATA #IMPLIED
            klasse CDATA #IMPLIED
            html-name CDATA #IMPLIED
        >
<!ELEMENT satz (mm+, anzahl?)>
<!ATTLIST satz kennung CDATA #IMPLIED
        >
<!ELEMENT wert (#PCDATA)>
<!ATTLIST wert einheit-basis (anzahl | k | m | m2 | m3 | text) "anzahl"
            einheit-faktor CDATA "1"
            einheit-text CDATA #IMPLIED
        >

<!-- Verarbeitungssteuerung
Es ist möglich, Testdokumente zu kennzeichnen und eine Empfangsbestä-
tigung anzufordern (Unterstützung vom Implementierungsgrad und ggf. von
der Erhebung abhängig)
-->
<!ELEMENT optionen (test?, empfangsbestaetigung?)>
<!ELEMENT test EMPTY>
<!ATTLIST test kennung CDATA #IMPLIED
        >
<!ELEMENT empfangsbestaetigung EMPTY>
<!ATTLIST empfangsbestaetigung versandart (default | email | post) "default"
            format (default | xml | text) "default"
            betreff CDATA #IMPLIED
        >

<!-- Protokollierung
Der Dateneingang und beliebig viele Dokumentinstanzen können proto-
kolliert werden.
-->
<!ELEMENT protokoll (eingang?, instanz*)>
<!ELEMENT eingang EMPTY>
<!ATTLIST eingang datum CDATA #IMPLIED
            uhrzeit CDATA #IMPLIED
            anwendung CDATA #IMPLIED
            liefermedium (undefiniert | internet | datentraeger) "undefiniert"
            datentraegertyp (undefiniert | diskette | cdrom | dvd | kassette |
                magnetband | fragebogen) "undefiniert"
        >

```

```
        datentraeger-id CDATA #IMPLIED
        lieferformattyp (undefiniert | datensatz | csv | xml | html) "undefiniert"
        lieferformatname CDATA #IMPLIED
        fragebogen CDATA #IMPLIED
        dateiname CDATA #IMPLIED
    >
<!ELEMENT instanz EMPTY>
<!ATTLIST instanz datum CDATA #IMPLIED
            uhrzeit CDATA #IMPLIED
            anwendung CDATA #IMPLIED
    >

<!-- Sonstige Elemente
-->
<!ELEMENT anzahl (#PCDATA)>
<!ELEMENT berechtigung (#PCDATA)>
<!ATTLIST berechtigung klasse CDATA "default"
    >
<!ELEMENT kennung (#PCDATA)>
<!ATTLIST kennung klasse CDATA "default"
    >
<!ELEMENT memo (#PCDATA)>
<!ELEMENT name (#PCDATA)>
<!ATTLIST name zeilennummer CDATA #IMPLIED
            html-name CDATA #IMPLIED
    >
<!ELEMENT string (#PCDATA)>
<!ATTLIST string format CDATA #REQUIRED
            klasse (undefiniert | datum) #REQUIRED
    >
<!ELEMENT text (#PCDATA)>
<!ELEMENT version (#PCDATA)>
<!ELEMENT zusatz (#PCDATA)>

<!-- Ende der DTD -->
```