

IT-Standards für die amtliche Statistik

Data Markup Language (DatML)

Komponente DatML/SDF 1.0

Ein XML-basierter Dokumenttyp zur Unterstützung der automatischen Generierung statistischer Rohdatennachrichten

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

Status: Spezifikation

Stand: 08.03.2005

© Statistisches Bundesamt Wiesbaden, Deutschland

Data Markup Language/SDF (DatML/SDF) 1.0

Spezifikation

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

Stand: 08.03.2005

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 ist auf die Bedürfnisse der Berichtspflichtigen ausgerichtet, die statistische Daten in XML liefern wollen, und 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/SDF für die Beschreibung statistischer Erhebungen, d.h. die Beschreibung der statistischen Erhebungsmerkmale und anderer erhebungsrelevanter Daten, insbesondere zum Zweck der Erzeugung und Prüfung von Datenlieferungen im Format DatML/RAW.
- 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/SDF beschrieben.

Dokumentstatus

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

Version	Datum	Autor	Stand/Änderung
1	08.03.2005	Schäfer	Spezifikation

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

1. Inhaltsverzeichnis

1. Inhaltsverzeichnis	4
2. Einführung	6
2.1. Allgemeines	6
2.2. Verwendung	8
2.2.1. Ein generischer Dokumenttyp	8
2.2.2. Anwendungen von DatML/SDF	8
3. Anforderungen	9
3.1. Fachliche Anforderungen	9
3.1.1. Vollständigkeit	9
3.1.2. Korrektheit	9
3.2. Organisatorische Anforderungen	10
3.2.1. Identifizierung	10
3.2.2. Erhebung	11
3.2.3. Merkmalsarten	11
3.2.4. Pünktlichkeit	11
3.2.5. Zusatzfunktionen	12
3.3. Technische Anforderungen	12
4. Logische Komponenten	13
4.1. Erhebung	13
4.2. Merkmale	14
4.3. Klassifikationen	16
5. Struktur und Eigenschaften von DatML/SDF	17
5.1. Struktur	17
5.1.1. Allgemeines	17
5.1.2. Aus DatML/SSP übernommene Strukturen	17
5.1.3. Allgemeines Datenmodell	17
5.1.4. Ausgabedatenmodell	17
5.1.5. Referenzdaten	18
5.1.6. Utilities	18
5.1.6.1. Parameter	18
5.1.6.2. Filter und Adapter	18
5.1.7. Merkmale in Ausgabegruppen inkludieren	19
5.2. Eigenschaften	19
5.2.1. Namen	19
5.2.1.1. Wertebereich	19
5.2.1.2. Namenskonflikte	19
5.3. Namensraum	19
6. Darstellung der Elemente	20
6.1. Darstellung der Elementtypen und Attribute	20
6.1.1. Graphische Darstellung der Elementtypen	20
6.1.2. Tabellarische Darstellung der Elementtypen	21
6.1.3. Tabellarische Darstellung der Attribute	21
6.2. Glossar	22
7. Elemente von DatML/SDF	23
7.1. Erhebung (Top-Level)	23
7.1.1. Element <code><sdf:surveyDefinition></code>	23
7.2. Datenmodell	24
7.2.1. Element <code><sdf:dataModel></code>	24
7.2.2. Komponenten Merkmal und Merkmalsgruppe	25

7.2.2.1. Element <sdf:variables>	25
7.3. Ausgabedatenmodell	26
7.3.1. Element <sdf:outputDataModel>	26
7.3.2. Komponente Ausgabegruppe	27
7.3.2.1. Element <sdf:group>	27
7.3.3. Komponente Ausgabefelder	28
7.3.3.1. Element <sdf:includeVariable>	28
7.3.3.2. Element <sdf:setValue>	29
7.3.3.3. Element <sdf:includeVariableGroup>	30
7.4. Referenzdaten / Klassifikation	31
7.4.1. Element <sdf:referenceData>	31
7.5. Utilities	32
7.5.1. Element <sdf:utilities>	32
7.5.2. Element <sdf:parameters>	33
7.5.3. Element <sdf:adapters>	34
7.5.4. Element <sdf:adapter>	35
7.5.5. Element <sdf:filters>	36
7.5.6. Element <sdf:filter>	37
8. In DatML/SDF verwendete Elemente von DatML/SSP	38
8.1. Properties	38
8.1.1. Element <ssp:properties>	38
8.2. Objektidentifikation und –referenz	39
8.2.1. Element <ssp:objectContext>	40
8.3. Kontext	42
8.3.1. Element <ssp:context>	42
8.4. Daten	43
8.4.1. Element <ssp:variable>	43
8.4.2. Element <ssp:simpleType>	44
8.4.3. Element <ssp:valueSpace>	45
8.4.4. Element <ssp:enumeration>	46
8.4.5. Element <ssp:enumElement>	47
8.4.6. Element <ssp:variableGroup>	48
8.4.7. Element <ssp:includeVariable>	49
8.4.8. Element <ssp:includeVariableGroup>	50
8.5. Klassifikation	51
8.5.1. Element <ssp:classification>	51
8.5.2. Element <ssp:classificationElement>	52
8.6. Basiselementtypen	53
8.6.1. Element <ssp:annotation>	53
8.6.2. Element <ssp:description>	54
9. Beispieldokument [nicht-normativ]	55
9.1. Beispiel 1	55
9.2. Beispiel 2	58
9.3. Beispiel 3	61
10. XML-Schema 1.0 [nicht-normativ]	72

2. Einführung

2.1. Allgemeines

Die statistischen Landesämter und das Statistische Bundesamt haben gemeinsam den XML-basierten Dokumenttyp DatML/RAW für die Übermittlung von Rohdaten an statistische Ämter entwickelt. Ihr Ziel ist es, DatML/RAW als einheitliches Schnittstellenformat und darauf aufbauend einen durchgehenden, generischen Datenerhebungsprozess mit einheitlichen Funktionen für alle Erhebungen zu implementieren. Dieser ist wiederum eine notwendige Vorstufe für weitere generische, metadatengestützte Verarbeitungsprozesse in den Fachverfahren und den Publikations- und Bereitstellungsprozessen.

Während damit einerseits die Leistungsfähigkeit und die Effizienz statistischer Verarbeitungsprozesse sowie die Bandbreite und Qualität von – insbesondere internetbasierten – nutzerorientierten Diensten erhöht werden können, sind andererseits erhöhte Aufwände bei der Erzeugung und Verarbeitung der komplexen Dokumentformate zu berücksichtigen. Dies trifft vor allem die Melder, da sie sich im Regelfall nicht auf eine breite Metadatenbasis und generische Prozesse stützen können. Als Folge sind die Implementierungs- und Wartungsaufwände schnell um ein Vielfaches höher als für herkömmliche Dateiformate, ohne dass ein entsprechender Nutzen entsteht.

In der bisherigen Praxis hat dies z.B. zur Bildung von Entwicklerkonsortien geführt, welche sich die Implementierungsaufwände für einzelne Erhebungen teilen. Hier handelt es sich aber immer noch um erhebungsspezifische, nicht-generische Lösungen, die zudem einen noch immer relativ hohen Wartungsbedarf haben.

Ein weiterer Fortschritt ist hier nur von generischen Verfahren zu erwarten, die es auch erlauben, einheitliche Softwarekomponenten zu entwickeln und für viele Erhebungen wiederzuverwenden. Solche Komponenten eignen sich z.B. für die Integration in betriebswirtschaftliche Anwendungen. Die Verwendung standardisierter, geprüfter Softwarekomponenten ist auch ein wichtiger Aspekt der Qualitätssicherung.

Die generische Erzeugung von Rohdatenmeldungen bedarf wie jeder andere generische Prozess externer Metadaten. An diesem Punkt setzt die Idee an, den Meldern die für die generische Erzeugung von Rohdatenmeldungen benötigten Metadaten mit Hilfe XML-basierter sogenannter *Erhebungsbeschreibungen* verfügbar zu machen. Erhebungsbeschreibungen sind *formale, maschinell auswertbare Beschreibungen statistischer Erhebungen*. Ziel ist es, Anwendungen mit Hilfe von Erhebungsbeschreibungen so zu parametrisieren, dass sie valide Rohdatennachrichten im Format DatML/RAW erzeugen.

Da der initiale Aufwand für die Implementierung generischer Lösungen im allgemeinen höher ist als für spezifische, lohnt er sich nur bei einem entsprechend großen Nutzerkreis. Dies ist bei Standardsoftware meist der Fall. Viele Melder verwenden

jedoch von ihnen selbst oder von Dritten entwickelte maßgeschneiderte Systeme, und oft sind sie bei Wartung und Entwicklung auf externe Dienste angewiesen. Hier ist es denkbar, eine durch Erhebungsbeschreibungen gesteuerte Anwendung bereitzustellen, die es den Meldern ermöglicht, aus ihren Anwendungssystemen exportierte Daten auf die Datenmodelle der statistischen Erhebungen abzubilden und generisch Rohdatennachrichten zu erzeugen.

2.2. Verwendung

2.2.1. Ein generischer Dokumenttyp

DatML/SDF (Data Markup Language - Survey Definition Format) unterstützt mit seiner umfassenden formalen Beschreibung der Eigenschaften und Merkmale einer Erhebung die Automatisierung der Datengewinnungsprozesse in der Unternehmens-DV und die Erzeugung der Lieferelemente. Der Einsatz effizienter generischer Methoden erleichtert dabei die Aktualisierung der Verfahren und gewährleistet eine hohe Datenqualität.

DatML/SDF beschreibt Erhebungen und die zu liefernden Merkmale so, dass vollständige und valide Rohdatenlieferungen einmalig oder über längere Zeiträume automatisch erzeugt werden können. Verbraucher von DatML/SDF sind z.B. Anwendungen des betrieblichen Rechnungswesens.

2.2.2. Anwendungen von DatML/SDF

Die primären Anwendungsfelder von DatML/SDF sind:

- Steuerung generischer Anwendungen, welche DatML/RAW-Dokumente erzeugen (Melderseite),
- Steuerung generischer Anwendungen, die DatML/RAW-Dokumente prüfen, verteilen und die in ihnen enthaltenen statistischen Daten und Metadaten (in ggf. anderer Form, z.B. als flache Datei) den Fachverfahren zuführen,
- Generierung erhebungsspezifischer Software, insbesondere von APIs, die das Datenmodell einer Erhebung abbilden.

3. Anforderungen

Die Anforderungen an Erhebungsbeschreibungen lassen sich in fachliche (d.h. statistische, insbesondere an die Qualität der Daten gerichtete), organisatorische und technische Anforderungen unterteilen. Diese Einteilung ist zwar grob, macht es aber leichter, sich den Anforderungen an eine Erhebungsbeschreibung systematisch zu nähern. Die technischen Anforderungen folgen überwiegend aus den Überlegungen, wie die fachlichen und organisatorischen Anforderungen durch Erhebungsbeschreibungen und den auf sie aufbauenden Methoden und Verfahren befriedigt werden können.

3.1. Fachliche Anforderungen

Die Gesamtheit der fachlichen Anforderungen an eine Erhebungsbeschreibung ergibt sich aus dem Ziel, statistische Daten für alle (relevanten) Erhebungsmerkmale (im weiteren nur *Merkmale* genannt) und Erhebungseinheiten formal und inhaltlich korrekt zu melden. Grundsätzliche fachliche Forderungen an statistische Rohdatenmeldungen sind daher *Vollständigkeit* und *Korrektheit*. Nicht berücksichtigt in der fachlichen Betrachtung der Merkmale ist, wie diese im Fachverfahren verwendet werden, also z.B. als Ordnungs- oder Wertmerkmal; dieser Aspekt wird, ebenso wie die auch aus fachlicher Sicht wichtige Pünktlichkeit unter den organisatorischen Anforderungen behandelt (s. 3.2).

3.1.1. Vollständigkeit

Der Aspekt Vollständigkeit wird an dieser Stelle mit Blick auf einzelne Erhebungseinheiten bzw. statistische Meldungen bewertet. Vollständigkeit in diesem Sinne verlangt, dass in einer statistischen Meldung mindestens für alle obligatorischen Merkmale Werte geliefert werden. Hierfür ist es sinnvoll, zwischen obligatorischen und freiwilligen Angaben zu unterscheiden. Allerdings ist dies nicht unbedingt eine fixe Größe, sie kann z.B. vom Wert eines anderen Merkmals abhängen.

Um eine vollständige Datenlieferung zu ermöglichen, muss eine Erhebungsbeschreibung alle Merkmale, ihre Optionalität und die zwischen ihnen bestehenden Abhängigkeiten beschreiben.

In statistischen Erhebungsverfahren ist es manchmal zulässig, Vollständigkeit durch mehrere Teillieferungen herzustellen. Die aktuelle Version von DatML/SDF behandelt diesen Aspekt jedoch noch nicht.

3.1.2. Korrektheit

Auf Ebene der Rohdaten beschränkt sich Korrektheit im wesentlichen auf die Konformität eines Datums mit dem verwendeten Datentyp und die Einhaltung des zulässigen Wertebereiches. Bei einem abgeleiteten Merkmal muss außerdem die Ablei-

tung – i.d.R. eine Berechnung – korrekt erfolgen. Die vollständige Plausibilität statistischer Daten ist erfahrungsgemäß erst innerhalb des Fachverfahrens überprüfbar.

Die Erhebungsbeschreibung muss den Datentyp und ggf. den gegenüber diesem eingeschränkten Wertebereich eines jeden Merkmals liefern. Klassifikationen können ggf. außerhalb der Erhebungsbeschreibung gespeichert sein. Es können Regeln für die Erzeugung bzw. Prüfung von Werten, etwa in der Form von Regular Expressions, beschrieben werden.

3.2. Organisatorische Anforderungen

Um einen durchgehenden generischen Datenerhebungsprozess vom Eingang der Meldung bis zur Übergabe der Rohdaten an das Fachverfahren zu implementieren, bedarf es einer entsprechenden Ausstattung der Rohdatenmeldungen mit organisatorischen Metadaten. Als wichtigste Bereiche sind hier zu nennen:

- Identifizierung der Teilnehmer (Melder, Amt,...),
- Zuordnung zur einer Erhebung,
- Art und Verwendung von Merkmalen,
- Überprüfung der Pünktlichkeit einer Lieferung und
- Zusatzfunktionen (Testmeldung, Adreßkorrektur,...)

3.2.1. Identifizierung

An einem Meldevorgang sind in jedem Fall ein Berichtspflichtiger und das zuständige statistische Amt beteiligt. Beide können durch Dritte vertreten werden (Mandantenschaft), so dass maximal vier Teilnehmer zu identifizieren sind.

I.d.R. sollten Daten, die Verfahrensbeteiligte, insbesondere den Berichtspflichtigen oder in seinem Namen handelnde Dritte, identifizieren, nicht Bestandteil einer Erhebungsbeschreibung sein. Die Gründe sind vor allem a) Datenschutz und b) Aufwand. Da es sich um persönliche Angaben handelt, müßte sichergestellt sein, dass eine Erhebungsbeschreibung mit solchen Informationen nur dem tatsächlichen Berichtspflichtigen bzw. dessen Vertreter zur Verfügung steht. Dies wäre am einfachsten und sichersten zu realisieren, indem eine solche Datei in einem abgesicherten Online-Verfahren zeitnah generiert und vom Anwender selbst heruntergeladen würde. In Frage käme evtl. auch eine automatische Verteilung durch den Versand an vom den Anwendern angegebene E-Mail-Adressen. In jedem Fall ist der Aufwand für die statistischen Ämter höher als bei einmaliger Generierung einer Erhebungsbeschreibung, die zentral zum Download zur Verfügung gestellt wird. Außerdem würden persönliche Daten übertragen, die auf der Anwenderseite zudem überprüft werden müßten.

Praktikabler und sinnvoller kann es sein, den Adressaten einer Meldung, also das statistische Amt bzw. eine dritte in dessen Namen handelnde Stelle in der Erhebungsbeschreibung einzutragen. Diese Information kann in die Meldungsdatei über-

nommen werden und damit eine korrekte Adressierung sicherstellen. Probleme können entstehen, wenn es mehrere potentielle Empfänger – z.B. Landesämter – gibt und ggf. eine Zuordnung zu einem davon, etwa über die Postleitzahl, erfolgen muss.

Grundsätzlich sollte es vermieden werden, Angaben zur Identifikation eines Teilnehmers in die Erhebungsbeschreibung aufzunehmen. Die aktuelle Version von DatML/RAW bietet aus diesem Grund keine Unterstützung für die Definition von Verfahrensbeteiligten.

3.2.2. Erhebung

Die Erhebung selbst muss selbstverständlich aus einer Erhebungsbeschreibung hervorgehen. Zur Erhebung gehören noch weitere Informationen, meist für dokumentarische Zwecke: textuelle Beschreibung, erhebende Institution, fachliche Zuständigkeit, Rechtsgrundlage, Berichtskreis, Auskunftspflicht, Berichtsweg, Umfang der Erhebung usw.

3.2.3. Merkmalsarten

Es gibt drei Arten von Merkmalen: Ordnungsmerkmale, Hilfs- und Wertmerkmale. Ordnungsmerkmale dienen i.d.R. zur Sortierung und Aggregation statistischer Objekte. Hilfsmerkmale bezeichnen oft Angaben, welche die Bearbeitung, Analyse und Plausibilisierung der Daten erleichtern. Die „eigentlichen“ statistischen Werte werden durch Wertmerkmale dargestellt. Oft besteht zwischen Ordnungs- und Hilfsmerkmalen auf der einen und den Wertmerkmalen auf der anderen Seite eine 1:n-Beziehung, so dass nicht alle Merkmale gleich häufig geliefert werden müssen. Die Differenzierung der Merkmalsarten in der Erhebungsbeschreibung kann somit zur Redundanzverringerung eingesetzt werden. Außerdem kann die Merkmalsart in generischen Sortierungs- und Aggregierungsschritten im späteren Fachverfahren verwendet werden.

Merkmalstypen sind vor allen in den statistischen Fachverfahren von Bedeutung, weniger für die Erzeugung von statischen Meldungen. DatML/SDF verzichtet auf einen explizite Merkmalstypen; stattdessen werden Merkmale indirekt im Ausgabedatenmodell durch Zuordnung zur Meldungs- bzw. Datensatzebene typisiert.

3.2.4. Pünktlichkeit

Erhebungsbeschreibungen können die Pünktlichkeit statistischer Meldungen durch die Angabe von Berichtszeitraum und Lieferfrist unterstützen. Bei periodischen Erhebungen können außerdem Liefertermine mit Hilfe der Periodizität berechnet werden. Damit entfällt auch die Notwendigkeit, eine Erhebungsbeschreibung für jeden Berichtstermin erneut zu beschaffen und auszuwerten. Allerdings muss dann dafür Sorge getragen werden, dass Erhebungsbeschreibungen im Falle von Änderungen beim Anwender aktualisiert werden, z.B. durch ein Verfallsdatum.

3.2.5. Zusatzfunktionen

Zusatzfunktionen im Sinne dieser Beschreibung sind Funktionen, die für eine Erhebung nicht unmittelbar benötigt werden oder als davon unabhängiger, separater Dienst zu sehen sind. Beispiele sind Testmeldungen, Rückantworten und Adresskorrekturen. Es ist möglich, dass solche Zusatzfunktionen erhebungsabhängig sind, d.h. nicht für alle Erhebungen zur Verfügung stehen, gewollt oder sinnvoll sind. Eine erhebungsbeschreibung sollte daher erlauben, die Zulässigkeit oder Notwendigkeit solcher Funktionen anzuzeigen.

3.3. Technische Anforderungen

Allgemeine technische Anforderungen wie Generizität, Lesbarkeit (Textbasiertheit), die Eignung für internetbasierte Verfahren und die allgemeine maschinelle Verarbeitbarkeit sollen hier nicht näher behandelt werden, da sie durch die Verwendung des XML-Standards als erfüllt gelten können.

Die folgenden Anforderungen stehen a priori fest:

- Aus einer Erhebungsbeschreibung muss ein DatML/RAW-Dokument generiert werden können; für die Nutzbarmachung aller Strukturierungsmöglichkeiten von DatML/RAW können spezielle Zusatzinformationen in der Erhebungsbeschreibung erforderlich sein. Grundsätzlich sollte das Datenmodell der Erhebungsbeschreibung aber möglichst unabhängig von dem der Rohdatennachricht sein.
- Eine Erhebungsbeschreibung muss versionierbar und ihre Gültigkeit für den aktuellen Berichtszeitraum muss maschinell ermittelbar sein.
- Eine Erhebungsbeschreibung muss eindeutig identifizierbar sein, damit ihre Verwendung kontrolliert werden kann.

Weitere besondere technische Anforderungen ergeben sich ggf. aus dem Datenmodell selbst bzw. aus der erhebungsspezifischen Beschreibung des zu liefernden DatML/RAW-Dokumentes (sog. Liefervereinbarung).

4. Logische Komponenten

Eine Erhebungsbeschreibung besteht aus einer Reihe einzeln beschreibbarer logischer Komponenten, die im Kontext einer Erhebung in bestimmten Beziehungen zueinander stehen. Aus diesen Bestandteilen – Komponenten und Beziehungen – wird das Datenmodell der Erhebungsbeschreibung entwickelt. Beziehungen werden in diesem Modell durch eine strukturierte Anordnung der Komponenten und durch Verweise ausgedrückt, wobei letztere ggf. auf externe Komponenten zeigen können.

Ausgangspunkt ist die Annahme, dass zum Mindestumfang einer Erhebungsbeschreibung zwei Komponenten gehören: die *Erhebung* selbst und deren *Merkmale*. Alle anderen Komponenten sind Nebenkomponten und meist einer der Hauptkomponenten untergeordnet.

4.1. Erhebung

Die Hauptkomponente Erhebung muss in einer Erhebungsbeschreibung genau einmal vorkommen und enthält mindestens die Informationen, notwendig sind, um auf maschinelle Weise die Erhebung zu identifizieren und einen Berichtszeitraum sowie einen Liefertermin festzustellen.

Die Tabelle zeigt die einzelnen Informationen, die zu einer Erhebung gehören. Die Optionalität einer Information ist mit **m** (muss vorhanden sein) und **k** (kann vorhanden sein) gekennzeichnet.

Eigenschaften von Erhebungen		
Identifikation	m	Eine Erhebung wird durch einen eindeutigen, formalen Schlüssel identifiziert.
Beschreibung	k	Eine textuelle Bezeichnung der Erhebung ist besonders für die Anzeige sinnvoll.
Berichtszeitraum	m	Der Berichtszeitraum ist der Zeitraum, für den statistische Daten über ein Erhebungsobjekt erhoben werden, typischerweise meist ein Jahr, ein Quartal oder ein Monat. Der Berichtszeitraum ist immer eine kalendarisch eindeutige, auf ein Jahr bezogene Angabe, bezeichnet also eine konkreten, einmaligen, nicht-periodischen Zeitraum. Eine Erhebungsbeschreibung kann einen initialen oder einen abschließenden Berichtszeitraum oder eine Liste zulässiger Berichtszeiträume enthalten.
Periodizität	k	Die Periodizität ist der Abstand zwischen zwei Erhebungen einer Statistik, also zwischen den Zeitpunkten, <i>an denen</i> Daten erhoben werden. Die Periodizität ist <i>nicht</i> der Ab-

		stand zwischen zwei Berichtszeiträumen.
Lieferfrist	k	Die Lieferfrist bestimmt den Zeitraum, innerhalb dessen nach Ende eines Berichtszeitraumes die statistischen Daten für diesen geliefert werden müssen.
Erhebende Stelle	k	Stelle, durch die die Daten erhoben werden, z.B. Bund, Land oder EU.
Zuständigkeit	k	Stelle, die fachlich für die Erhebung zuständig ist, d.h. die Erhebung fachlich betreut.
Rechtsgrundlagen	k	Die Rechtsgrundlage(n), auf der/denen die Erhebung beruht. Möglich ist ein Verweis auf eine Quelle, eine bloße Bezeichnung oder die Angabe des vollständigen Textes der Rechtsgrundlage(n).
Berichtskreis	k	Der Berichtskreis bezeichnet diejenigen natürlichen oder juristischen Personen, die Rohdaten für eine Erhebung liefern.
Erhebungsobjekte	k	Erhebungsobjekte sind die Objekte, über die statistische Daten erhoben werden, z.B. Haushalte, Unternehmen etc.
Auskunftspflicht	k	Gibt an, ob eine Auskunftspflicht besteht.
Organisationsform (Berichtsweg)	k	Gibt an, ob eine Erhebung zentral oder dezentral durchgeführt wird.

4.2. Merkmale

Die Tabelle zeigt die einzelnen Informationen, die zu einem Merkmal gehören. Die Optionalität einer Information ist mit **m** (muss vorhanden sein) und **k** (kann vorhanden sein) gekennzeichnet.

Eigenschaften von Merkmalen		
Name	m	Eine Merkmal wird durch einen (innerhalb der Erhebungsbeschreibung) eindeutigen Namen identifiziert.
Art des Merkmals	k	Die Art des Merkmals (Ordnungs-, Hilfs-, Wertmerkmal). Fehlt diese Angabe, wird ein Wertmerkmal angenommen.
Datentyp	m	Der Datentyp eines Merkmals. Der Datentyp legt die Syn-

		<p>tax einer Wertangabe fest: Ganzzahl, Dezimalzahl, Zeichenkette usw. Er enthält, abgesehen von Datums- und zeitformaten, keine implizite Beschränkung des Wertebereiches.</p>
Wertebereich	k	<p>Der Wertebereich bestimmt den Bereich zulässiger Werte eines Merkmals. Möglich sind (Listen von) Spannen und Einzelwerte, auch in Kombination, Enumerationstypen und die Verwendung von Klassifikationen. Es ist ggf. möglich, dass es für einen gegebenen zulässigen Wert mehr als eine zulässige lexikalische Ausprägung gibt, z.B. sind „100“ und „100,0“ verschiedene lexikalische Ausprägungen desselben dezimalen Wertes.</p>
Optionalität	k	<p>Die Optionalität legt fest, ob ein Wert für ein Merkmal geliefert werden muss oder ob die Angabe freiwillig ist. Fehlt die Angabe, muss ein Wert für das Merkmal geliefert werden. Optionalität kann jedoch abgeleitet oder kontextabhängig sein, d.h. sie kann z.B. von den Werten anderer Variablen abhängen.</p>
Spezifikationsattribute	k	<p>Die Spezifikationsattribute legen zusammen mit der Optionalität fest, ob und wie ein Wert für ein Merkmal geliefert werden muss. Es gibt vier Spezifikationsattribute:</p> <ul style="list-style-type: none"> - Leere Angabe, - Auskunftverweigerung, - Nichtwissen, - Wertüberschreitung <p>Die Spezifikationsattribute sind boolesche Operanden mit den Werten <i>erlaubt</i> und <i>nicht erlaubt</i>.</p>
Formel (Ableitung)	k	<p>Der Wert eines Wertmerkmals kann aus denen anderer Wertmerkmale abgeleitet sein. Beispiele sind die rechnerische Ableitung, wie Summenbildung und Prozentrechnung, und die Ableitung einer Gliederungsposition aus einem Wert. Bei abgeleiteten Merkmalen sind Optionalität und Art des Merkmals irrelevant bzw. festgelegt. Die Datentypen aller beteiligten Merkmale müssen miteinander kompatibel sein.</p>
Beschreibung	k	<p>Die textuelle Beschreibung eines Merkmals dient der Dokumentation und der Anzeige.</p>

4.3. Klassifikationen

In der amtlichen Statistik werden Klassifikationen meist für die Beschreibung und Gruppierung von Objekten der realen Welt mit Hilfe bestimmter Ordnungsschemata verwendet. Beispiele sind das *Einheitliche Verzeichnis der amtlichen Statistiken* (EVAS), die *Klassifikation der Wirtschaftszweige* (WZ 2003) und die *Güterliste für europäische Produktionsstatistiken* (PRODCOM).

Jede Klassifikation enthält eine beliebige Anzahl von *Elementen*. Zu jedem Element gehören eine *Beschreibung oder Bezeichnung* und ein eindeutiger *Schlüssel*. Schlüssel sind i.d.R. so strukturiert, dass sie Elemente und Bereiche von Elementen hierarchisch anordnen.

In einer statistischen Erhebung werden Klassifikationen meist dann benötigt, wenn einem Merkmal als Wert ein Schlüssel aus einer Klassifikation zugeordnet ist. In diesem Fall legt die Klassifikation den Wertebereich eines Merkmals fest.

Erhebungsbeschreibungen können Klassifikationen enthalten oder auf solche verweisen. DatML/SSP stellt Strukturen für die Beschreibung von Klassifikationen bereit, die ausführlich in der Spezifikation von DatML/SSP beschrieben sind.

5. Struktur und Eigenschaften von DatML/SDF

5.1. Struktur

5.1.1. Allgemeines

Jedes DatML/SDF-Dokument beschreibt eine Erhebung. Eine Erhebungsbeschreibung muss folgende den Erhebungskontext (`<ssp:context>`) und das Datenmodell (`<sdf:dataModel>`) – Merkmale und Merkmalsgruppen (`<ssp:variable>` und `<ssp:variableGroup>`) – enthalten. Darüberhinaus können weitere Angaben enthalten sein, unter anderem die Beschreibung eines Ausgabedatenmodells (`<sdf:outputDataModel>`), in dem die Merkmale und Merkmalsgruppen zu Ausgabegruppen (`<group>`) zusammengefasst; letzteres geschieht durch Referenzierung (`<sdf:includeVariable>` und `<sdf:includeVariableGroup>`).

DatML/SDF verwendet außer den eigenen Elementtypdefinitionen solche aus dem Namensraum von DatML/SSP. Letztere sind in der Regel Definitionen allgemein verwendbarer Strukturen, wie z.B. Dokumenteigenschaften (`<ssp:properties>`), während erstere die spezifischen Eigenschaften von DatML/SDF implementieren.

5.1.2. Aus DatML/SSP übernommene Strukturen

Aus dem Namensraum von DatML/SSP stammen folgende Elemente der Top-Level-Ebene:

- `<ssp:properties>`: Dokumenteigenschaften
- `<ssp:objectContext>`: Eigenschaften der Erhebungsbeschreibung als Gesamtobjekt.
- `<ssp:context>`: Fachlicher Kontext, in dem die Erhebungsbeschreibung verwendet wird, also Erhebung, Berichtszeitraum, usw.

5.1.3. Allgemeines Datenmodell

Das allgemeine Datenmodell (`<sdf:dataModel>`) der Erhebung umfasst in der aktuellen Version lediglich die Beschreibung der Erhebungsmerkmale (`<sdf:variables>`). Es ist geplant, das Datenmodell so zu erweitern, dass es auch Beziehungen zwischen Merkmalen darstellen kann, insbesondere hierarchische Beziehungen, z.B. in Form von Tabellen, und Ableitungen.

5.1.4. Ausgabedatenmodell

Das Ausgabedatenmodell (`<sdf:outputDataModel>`) hat den Zweck, die Merkmale in einer für die Erzeugung der Rohdatennachricht geeigneten Weise anzuordnen. Dies geschieht, indem Merkmale zu Gruppen (`<sdf:group>`) zusammengefasst und dann einer logischen Ebene des Ausgabedatenmodells zugeordnet werden. DatML/SDF unterstützt ein inhärentes Datenmodell mit drei logischen Ebenen: *Dokument*, *Meldung* und *Datensatz*. Dies entspricht dem Datenmodell von DatML/RAW. Auf Dokumentebene existieren in DatML/RAW keine Merkmale, und die Erzeugung der Dokumentebene ist durch die Spezifikation vollständig geregelt.

Die Dokumentenebene findet daher im Ausgabedatenmodell keine Berücksichtigung. Merkmale der Meldungsebene werden nur ein einziges Mal erzeugt und besitzen Gültigkeit für die ganze Meldung, solche der Datensatzebene können in jeder Instanz eines Datensatzes vorkommen.

HINWEIS: Das Ausgabedatenmodell ist optional, da es stets möglich ist, ein Ausgabedatenmodell ganz oder teilweise aus dem allgemeinen Datenmodell abzuleiten. Die einfachste Form der Ableitung ist die Erzeugung eines DatML/RAW-Datensatzes mit allen beschriebenen Merkmalen und Merkmalsgruppen. Die notwendige Einheitlichkeit der Verfahren erfordert die Definition von Ableitungsregeln für eine zukünftige Version von DatML/SDF. Solange diese nicht zur Verfügung stehen, sollten alle DatML/SDF-basierten Erhebungsbeschreibungen ein Ausgabedatenmodell enthalten.

5.1.5. Referenzdaten

Referenzdaten werden vor allem für die Evaluierung von Rohdaten benötigt. DatML/SDF unterstützt in dieser Version lediglich Klassifikationen.

5.1.6. Utilities

Utilities implementieren Zusatzfunktionen.

5.1.6.1. Parameter

Parameter übermitteln Steuerinformationen an Anwendungen, die Erhebungsbeschreibungen verarbeiten. Es ist vorgesehen, bei Bedarf entsprechende Parameter zu entwickeln, beispielsweise, um einer Anwendung die Zulässigkeit von Drittmeldern, die erlaubten Zeichensätze usw. mitzuteilen. Die Parameter sind dabei hinsichtlich ihrer Eigenschaften wie Name und Werte abschließend festgelegt und richten sich nicht an eine spezifische Anwendung.

Das Element `<sdf:parameters>` hat gegenwärtig nur eine Platzhalterfunktion.

5.1.6.2. Filter und Adapter

Filter (`<sdf:filters>`) und Adapter (`<sdf:adapters>`) implementieren zusammen die *anwendungsgesteuerte* Auswahl von Merkmalen und Ausgabegruppen. Zweck ist beispielsweise die Anwendung einer Benutzermatrix auf die zu liefernden Merkmale und Ausgabegruppen.

Ein Adapter hat einen Namen, einen Datentyp und einen Wertebereich. Die Anwendung versorgt den Adapter zu Laufzeit mit einem Wert, der mit dem Datentyp des Adapters übereinstimmen und in dessen Wertebereich liegen muss.

Ein Filter hat einen Namen und einen Wertebereich und referenziert einen Adapter. Bei der Anwendung eines Filters wird dieser mit dem referenzierten Adapter verglichen und als Ergebnis einer der booleschen Werte WAHR und FALSCH geliefert. Der Vergleich liefert den Wert WAHR, wenn der aktuelle Wert des Adapters im Wer-

tebereich des Filters liegt oder der Adapter keinen Wert hat, d.h. nicht von der Anwendung versorgt worden ist. In allen anderen Fällen liefert der Vergleich den Wert FALSCH.

Filter werden von Inkludierungsanweisungen (`<sdf:include*>`) und Ausgabegruppen (`<sdf:group>`) mit ihrem Namen referenziert und auf diese angewendet. Bei gleichzeitiger Referenzierung mehrerer Filter werden diese nacheinander evaluiert und so behandelt, als wären sie mit einem logischen UND verknüpft, d.h. das Ergebnis ist WAHR, wenn alle Filter den Wert WAHR zurückliefern.

5.1.7. Merkmale in Ausgabegruppen inkludieren

Eine nähere Beschreibung der allgemeinen Vorgehensweise beim Inkludieren von Merkmalen in Ausgabegruppen wird zu einem späteren Zeitpunkt ergänzt.

5.2. Eigenschaften

5.2.1. Namen

5.2.1.1. Wertebereich

Zahlreiche Elemente bzw. Komponenten von DatML/SDF werden durch 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.

5.2.1.2. Namenskonflikte

Namenskonflikte können auf zwei Ebenen auftreten: Innerhalb des SDF-Dokumentes und innerhalb der Rohdatennachricht. Letztere werden durch die jeweilige Nachrichtenspezifikation geregelt und hier nicht behandelt. Innerhalb DatML/SDF gilt, dass zwei Namen einen Namenskonflikt auslösen, wenn sie gleich sind und Objekte des gleichen Typs bezeichnen.

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.

5.3. Namensraum

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

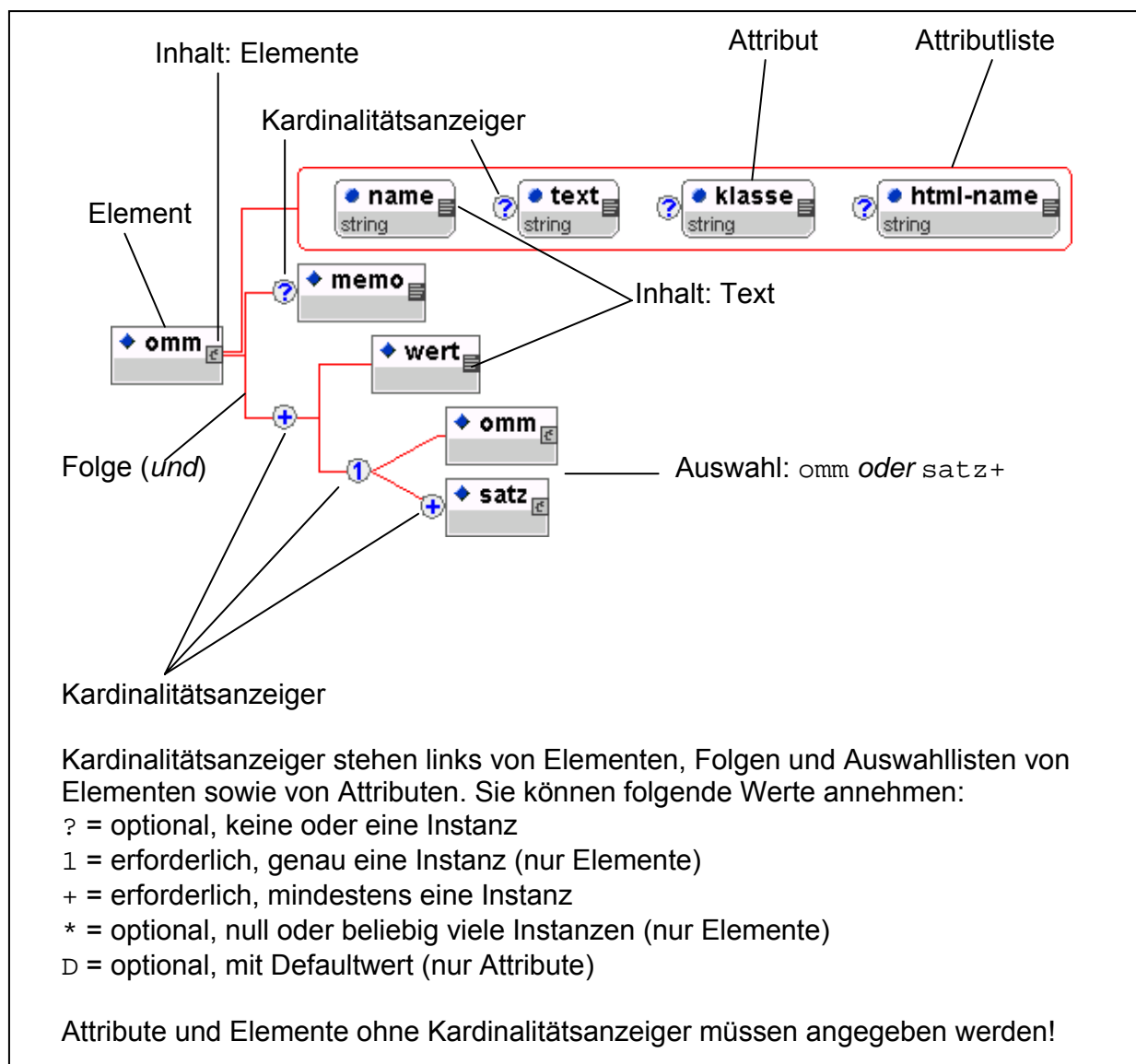
<http://www.destatis.de/schema/datml-sdf/1.0>.

6. Darstellung der Elemente

In den folgenden Kapiteln 7 und 8 werden die Elementtypen beschrieben, sie sind in ihrer Beschreibung unterteilt in Elemente von DatML/SDF und Elemente von DatML/SDF.

6.1. Darstellung der Elementtypen und Attribute

6.1.1. Graphische Darstellung der Elementtypen



6.1.2. Tabellarische Darstellung der Elementtypen

Element	<***Elementname***>		
Status	<i>status</i>	[<i>r</i>]	<i>version</i>
Datentyp:	<i>datentyp</i>		
Namespace:	http://www.destatis.de/schema/datml-raw/1.0/d	DTD:	-
Inhaltsmodell:	<i>inhaltsmodell</i>		
<i>graphik</i> (s. 6.1.1)			

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 '≤' und '≥' den Gültigkeitsraum mit einem einschließenden ‚ab‘ bzw. ‚bis‘ genauer eingrenzt.

6.1.3. 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

6.2. Glossar

Begriff	Erläuterung
Anwendung	Eine geeignete softwaretechnische Lösung, die diese Spezifikation implementiert.
Statistischer Kontext	Summe der Metadaten, die eine Einzelmeldung in einen für die statistische Verarbeitung relevanten Kontext stellt, insbesondere Erhebung und Berichtszeitraum.

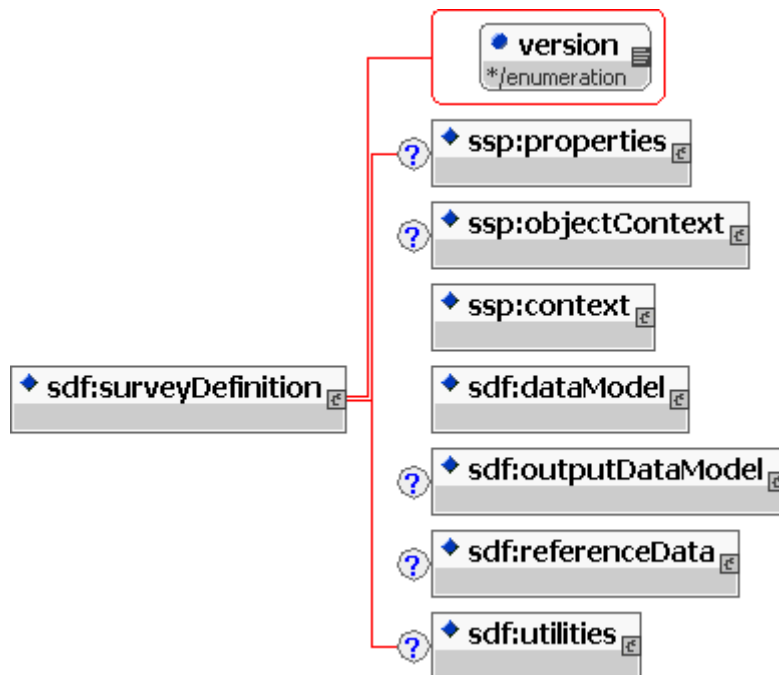
7. Elemente von DatML/SDF

7.1. Erhebung (Top-Level)

Ein DatML/SDF-Dokument enthält die Beschreibung einer Erhebung.

7.1.1. Element `<sdf:surveyDefinition>`

Element	<code><sdf:surveyDefinition></code>		
Datentyp:	<code>element-content</code>		
Namespace:	<code>http://www.destatis.de/schema/datml-sdf/1.0</code>		DTD: -



Attribute:	S	Datentyp, Default, Werte
version	M	enumeration:1.0; Versionsbezeichnung

Details

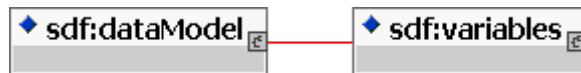
Eine Erhebungsbeschreibung (`<ssp:surveyDefinition>`) enthält mindestens den statistischen Verwendungskontext (`<ssp:context>`) sowie das Datenmodell (`<sdf:dataModel>`). Weitere Angaben wie die allgemeinen Dokumenteigenschaften (`<ssp:properties>`, 0), die Objektidentifikation (`<ssp:objectContext>`, 8.2.1), das Ausgabedatenmodell (`<sdf:outputDataModel>`, 7.3.1), Referenzdaten (`<sdf:referencedata>`, 7.4.1) und Zusatzfunktionen (`<sdf:utilities>`, 7.5.1) sind optional.

7.2. Datenmodell

Jede Erhebungsbeschreibung muss ein Datenmodell (`<sdf:dataModel>`) enthalten; dieses besteht aus Beschreibungen der Merkmale (`<ssp:variable>`) und der Merkmalsgruppen (`<ssp:variableGroup>`).

7.2.1. Element `<sdf:dataModel>`

Element	<code><sdf:dataModel></code>	
Datentyp:	<i>element-content</i>	
Namespace:	<code>http://www.destatis.de/schema/datml-sdf/1.0</code>	DTD: -



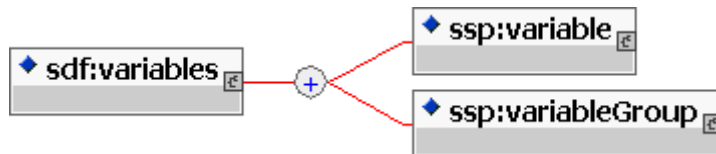
Details

Enthält die Beschreibungen der Merkmale und Merkmalsgruppen.

7.2.2. Komponenten Merkmal und Merkmalsgruppe

7.2.2.1. Element `<sdf:variables>`

Element	<code><sdf:variables></code>	
Datentyp:	<i>element-content</i>	
Namespace:	<code>http://www.destatis.de/schema/datml-sdf/1.0</code>	DTD: -



Details

Eine Folge von Definitionen von Merkmalen (`<ssp:variable>`, 8.4.1) und/oder Merkmalsgruppen (`<ssp:variableGroup>`, 8.4.6).

7.3. Ausgabedatenmodell

in einem Ausgabedatenmodell (`<sdf:outputDataModel>`) werden Merkmale und Merkmalsgruppen zu Ausgabegruppen (`<group>`) zusammengefasst; letzteres geschieht durch Referenzierung (`<sdf:includeVariable>` und `<sdf:includeVariableGroup>`). Jede Ausgabegruppe ist entweder der Meldungsbene oder der Datenbene des Ausgabedatenmodells zuzuordnen.

7.3.1. Element `<sdf:outputDataModel>`

Element	<code><sdf:outputDataModel></code>	
Datentyp:	<i>element-content</i>	
Namespace:	<code>http://www.destatis.de/schema/datml-sdf/1.0</code>	DTD: -



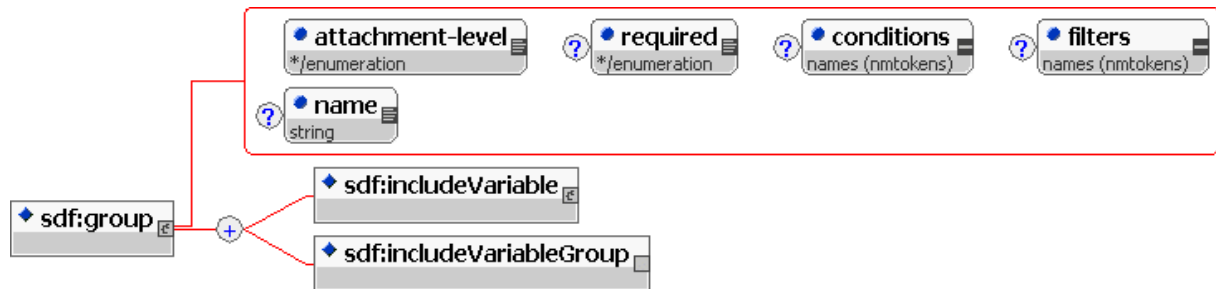
Details

Das Ausgabedatenmodell aus einer Folge von Ausgabegruppen.

7.3.2. Komponente Ausgabegruppe

7.3.2.1. Element `<sdf:group>`

Element	<code><sdf:group></code>		
Datentyp:	<i>element-content</i>		
Namespace:	<code>http://www.destatis.de/schema/datml-sdf/1.0</code>	DTD:	-



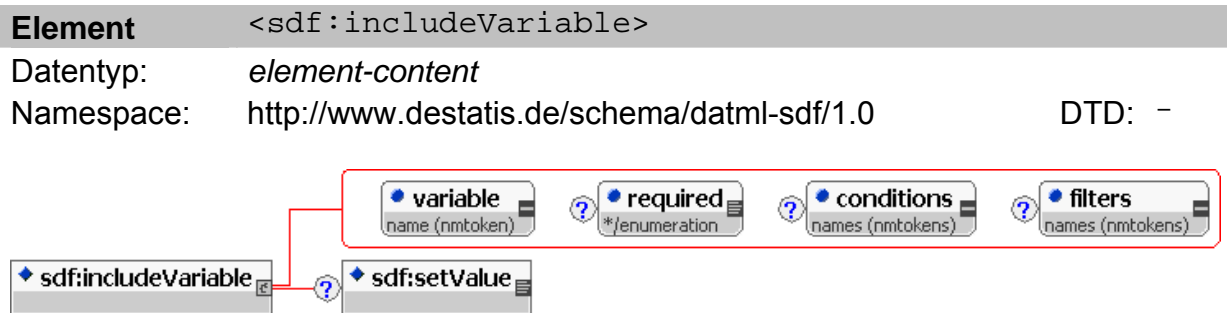
Attribute:	S	Datentyp, <u>Default</u> , Werte
attachment-level	M	<i>enumeration: data/message</i> ; Gibt an, ob es sich um Daten (Rohdaten) oder Metadaten handelt
required	?	<i>enumeration: yes/no/conditional</i> ; Gibt an, ob die Ausgabegruppe vorkommen muss
conditions	?	<i>char: Bedingung</i>
filters	?	<i>char: Filter</i>
name	?	<i>char: Name der Ausgabegruppe</i>

Details

Eine Ausgabedatenmodell besteht aus einer Folge von Referenzen auf einzelne Merkmale (`<sdf:includeVariable>`) oder von Referenzen auf Merkmalsgruppen (`<sdf:includeVariableGroup>`).

7.3.3. Komponente Ausgabefelder

7.3.3.1. Element `<sdf:includeVariable>`



Attribute:	S	Datentyp, <u>Default</u> , Werte
variable	M	<i>char</i> :Name des Merkmals
required	?	<i>enumeration</i> : <u>yes</u> /no/conditional; Gibt an, ob das Ausgabefeld vorkommen muss
conditions	?	<i>char</i> : Bedingung
filters	?	<i>char</i> : Filter

Details

Das Ausgabefeld `<sdf:includeVariable>` beschreibt eine Referenz auf ein Merkmal. Das Merkmal kann dabei mit einem ggf. unveränderlichen Wert vorbelegt werden.

Zur allgemeinen Vorgehensweise beim Inkludieren s. 5.1.7.

7.3.3.2. Element `<sdf:setValue>`

Element	<code><sdf:setValue></code>	
Datentyp:	<code>char</code>	
Namespace:	<code>http://www.destatis.de/schema/datml-sdf/1.0</code>	DTD: -



Attribute:	S	Datentyp, <u>Default</u>, Werte
<code>mutable</code>	D	<code>char:yes <u>no</u></code>

Details

Eine Referenz auf ein Merkmal kann mit einem Wert vorbelegt werden. Diese Wertvorgabe wird im Element `<sdf:setValue>` beschrieben und kann konstant sein.

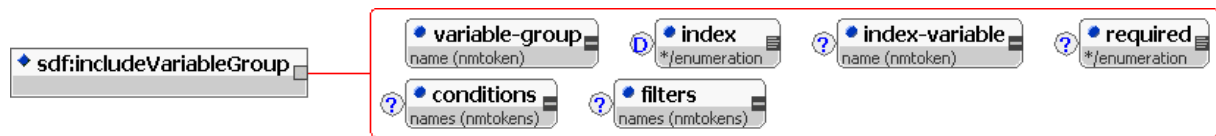
7.3.3.3. Element `<sdf:includeVariableGroup>`

Element `<sdf:includeVariableGroup>`

Datentyp: *element-content*

Namespace: `http://www.destatis.de/schema/datml-sdf/1.0`

DTD: -



Attribute: S Datentyp, Default, Werte

<code>variable-group</code>	M	<i>char</i> : Name der Merkmalsgruppe
<code>required</code>	?	<i>enumeration</i> : <u>yes</u> /no/conditional; Gibt an, ob das Ausgabefeld vorkommen muss
<code>conditions</code>	?	<i>char</i> : Bedingung
<code>filters</code>	?	<i>char</i> : Filter
<code>index</code>	D	<i>enumeration</i> : <u>automatic</u> /by-variable/undefined
<code>index-variable</code>	?	<i>char</i> :

Details

Das Ausgabefeld `<sdf:includeVariableGroup>` beschreibt eine Referenz auf eine Merkmalsgruppe.

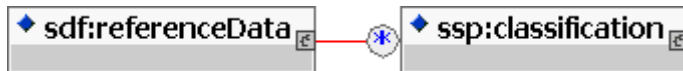
Zur allgemeinen Vorgehensweise beim Inkludieren s. 5.1.7.

7.4. Referenzdaten / Klassifikation

Diese Angabe ist optional in einer Erhebungsbeschreibung.

7.4.1. Element `<sdf:referenceData>`

Element	<code><sdf:referenceData></code>	
Datentyp:	<i>element-content</i>	
Namespace:	<code>http://www.destatis.de/schema/datml-sdf/1.0</code>	DTD: -



Details

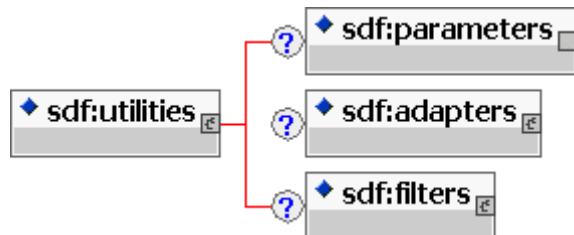
Enthält eine optionale Folge von Klassifikationen (8.5).

7.5. Utilities

Diese Angabe ist optional in einer Erhebungsbeschreibung und beschreibt Zusatzfunktionen.

7.5.1. Element `<sdf:utilities>`

Element	<code><sdf:utilities></code>	
Datentyp:	<i>element-content</i>	
Namespace:	<code>http://www.destatis.de/schema/datml-sdf/1.0</code>	DTD: -



Details

Beschreibt die Zusatzfunktionen; dies sind Parameterübergabe sowie Adapter und Filter, mit denen Auswahl schemata für Merkmale realisiert werden können.

7.5.2. Element `<sdf:parameters>`

Element	<code><sdf:parameters></code>	
Datentyp:	<i>empty-element</i>	
Namespace:	<code>http://www.destatis.de/schema/datml-sdf/1.0</code>	DTD: -



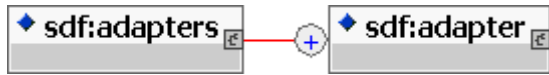
Details

Parameter übermitteln Steuerinformationen an Anwendungen, die Erhebungsbeschreibungen verarbeiten. Es ist vorgesehen, bei Bedarf entsprechende Parameter zu entwickeln, beispielsweise, um einer Anwendung die Zulässigkeit von Drittmeldern, die erlaubten Zeichensätze usw. mitzuteilen. Die Parameter sind dabei hinsichtlich ihrer Eigenschaften wie Name und Werte abschließend festgelegt und richten sich nicht an eine spezifische Anwendung.

Das Element `<sdf:parameters>` hat eine gegenwärtig nur Platzhalterfunktion.

7.5.3. Element `<sdf:adapters>`

Element	<code><sdf:adapters></code>	
Datentyp:	<i>element-content</i>	
Namespace:	<code>http://www.destatis.de/schema/datml-sdf/1.0</code>	DTD: -

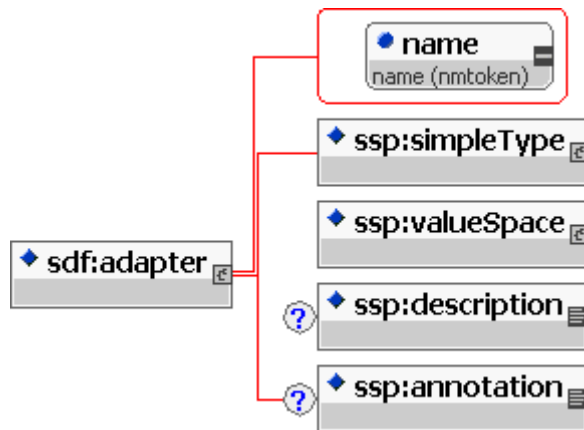


Details

Filter (`<sdf:filters>`) und Adapter (`<sdf:adapters>`) implementieren zusammen die *anwendungsgesteuerte* Auswahl von Merkmalen und Ausgabegruppen. Zweck ist beispielsweise die Anwendung einer Benutzermatrix auf die zu liefernden Merkmale und Ausgabegruppen.

7.5.4. Element <sdf:adapter>

Element	<sdf:adapter>	
Datentyp:	<i>element-content</i>	
Namespace:	http://www.destatis.de/schema/datml-sdf/1.0	DTD: -



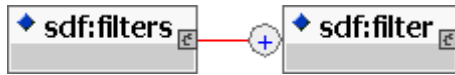
Attribute: **S** **Datentyp, Default, Werte**
 name M *char:*

Details

Ein Adapter hat einen Namen, einen Datentyp und einen Wertebereich. Die Anwendung versorgt den Adapter zu Laufzeit mit einem Wert, der mit dem Datentyp des Adapters übereinstimmen und in dessen Wertebereich liegen muss.

7.5.5. Element `<sdf:filters>`

Element	<code><sdf:filters></code>	
Datentyp:	<i>element-content</i>	
Namespace:	<code>http://www.destatis.de/schema/datml-sdf/1.0</code>	DTD: -

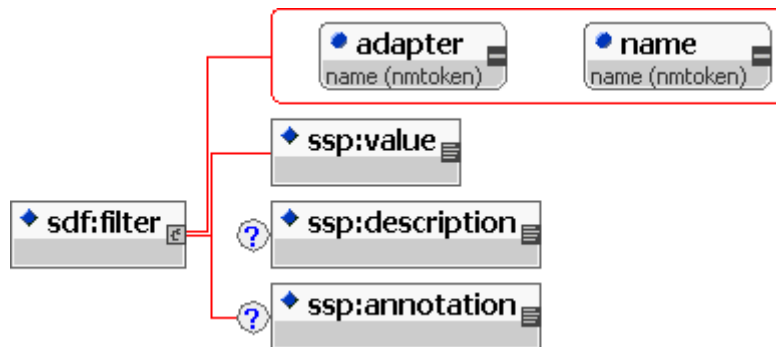


Details

Filter (`<sdf:filters>`) und Adapter (`<sdf:adapters>`) implementieren zusammen die *anwendungsgesteuerte* Auswahl von Merkmalen und Ausgabegruppen. Zweck ist beispielsweise die Anwendung einer Benutzermatrix auf die zu liefernden Merkmale und Ausgabegruppen.

7.5.6. Element `<sdf:filter>`

Element	<code><sdf:filter></code>	
Datentyp:	<i>element-content</i>	
Namespace:	<code>http://www.destatis.de/schema/datml-sdf/1.0</code>	DTD: -



Attribute:	S	Datentyp,	Default,	Werte
<code>adapter</code>	M	<i>char:</i>		
<code>name</code>	M	<i>char:</i>		

Details

Ein Filter hat einen Namen und einen Wertebereich und referenziert einen Adapter. Bei der Anwendung eines Filters wird dieser mit dem referenzierten Adapter verglichen und als Ergebnis einer der booleschen Werte WAHR und FALSCH geliefert. Der Vergleich liefert den Wert WAHR, wenn der aktuelle Wert des Adapters im Wertebereich des Filters liegt oder der Adapter keinen Wert hat, d.h. nicht von der Anwendung versorgt worden ist. In allen anderen Fällen liefert der Vergleich den Wert FALSCH.

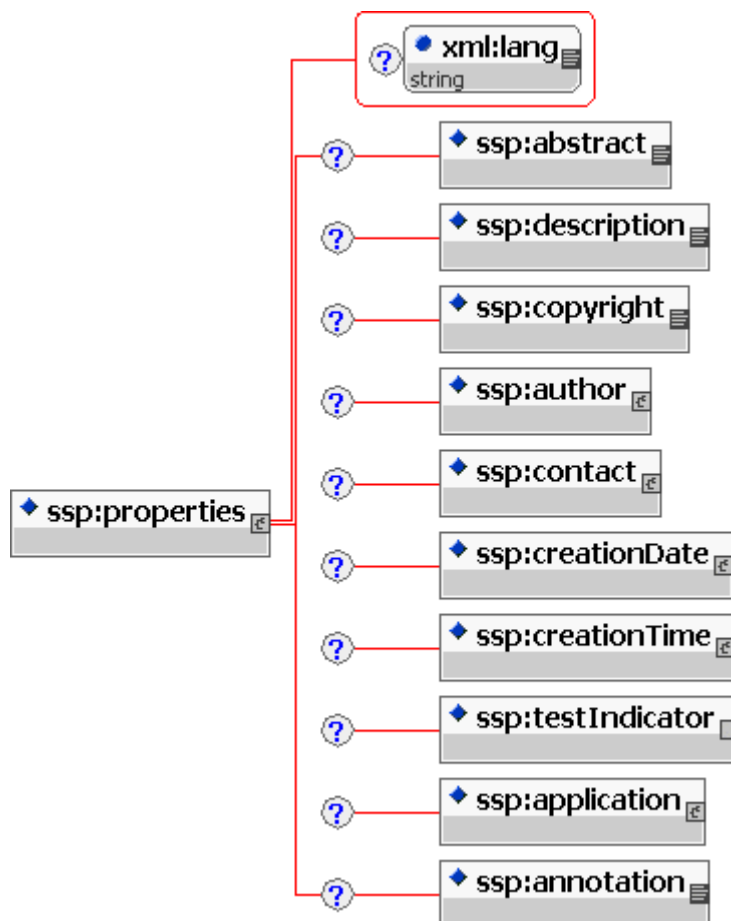
8. In DatML/SDF verwendete Elemente von DatML/SSP

8.1. Properties

In einer Erhebung können optionale properties-Angaben, d.h. allgemeine dokumentbezogene Metadaten wie Autor, Datum usw. abgelegt werden.

8.1.1. Element `<ssp:properties>`

Element	<code><ssp:properties></code>		
Datentyp:	<code>element-content</code>		
Namespace:	<code>http://www.destatis.de/schema/datml-ssp/1.0/de</code>	DTD:	-



Attribute:	S Datentyp, Default, Werte
<code>xml:lang</code>	? <code>char</code> : Sprache

Details

Das optionale Element `<ssp:properties>` dient zur Ablage allgemeiner dokumentbezogener Metadaten wie Autor, Datum usw.

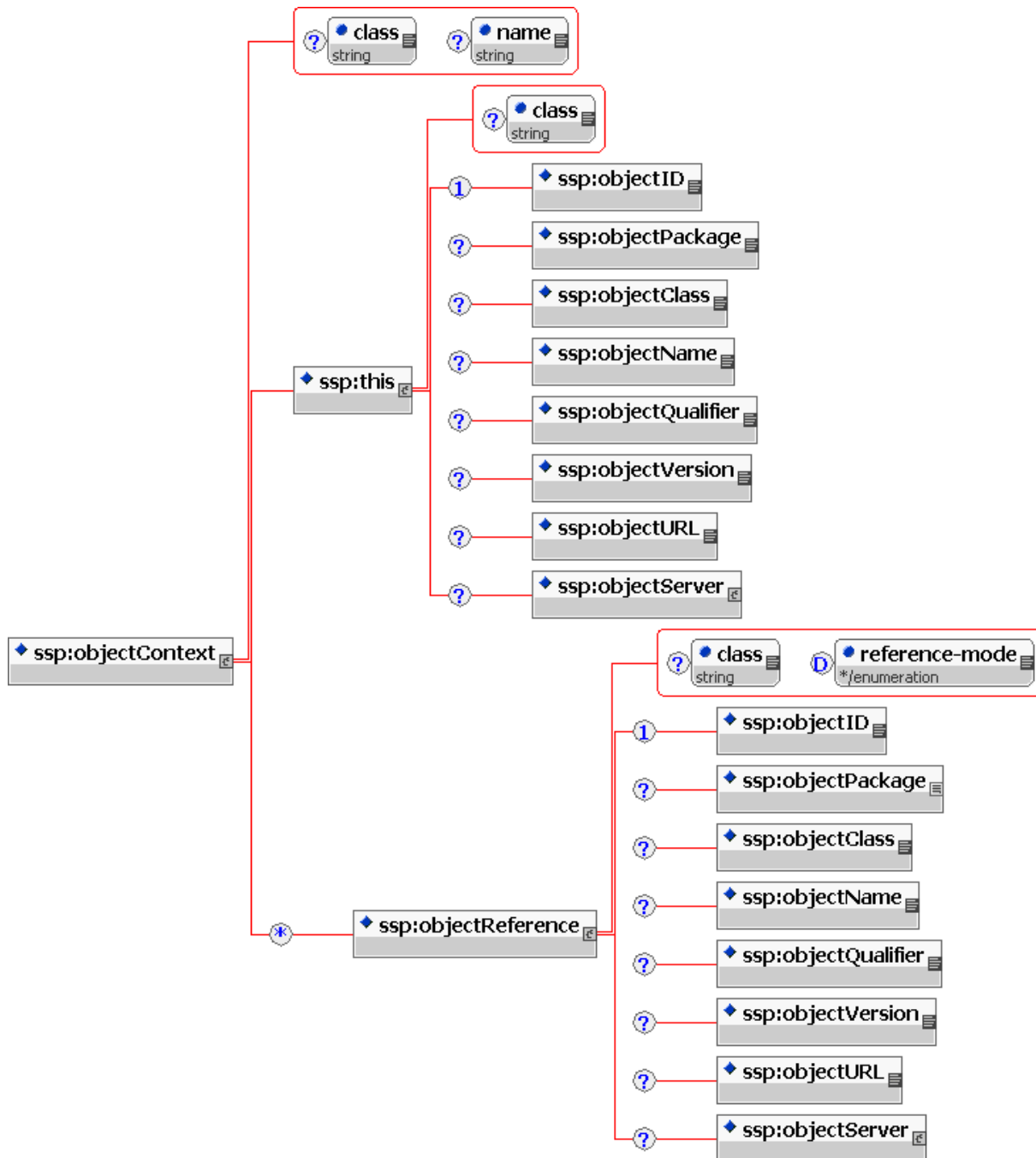
8.2. Objektidentifikation und –referenz

Diese Angabe ist optional in einer Erhebungsbeschreibung.

Objekte wie Datensatzbeschreibungen, Merkmalsdefinitionen usw. haben i.d.R. einen persistenten Speicher, z.B. eine Datenbank oder ein Filesystem. Sehr häufig bestehen Beziehungen zwischen Objekten, z.B. aufgrund von Ableitung oder eines fachlichen Zusammenhangs. Der Elementtyp `<ssp:objectContext>` bietet Strukturen, die entsprechende Eigenschaften des aktuellen (`<this>`) und auf dieses bezogener Objekte (`<objectReference>`) beschreiben.

8.2.1. Element <ssp:objectContext>

Element	<ssp:objectContext>		
Datentyp:	element-content		
Namespace:	http://www.destatis.de/schema/datml-ssp/1.0/de	DTD:	-



Attribute:	S	Datentyp, Default, Werte
class	?	char: Kontextklasse
name	?	char: Kontextname
reference-mode	D	enumeration: <u>undefined</u> source target bidirectional ;

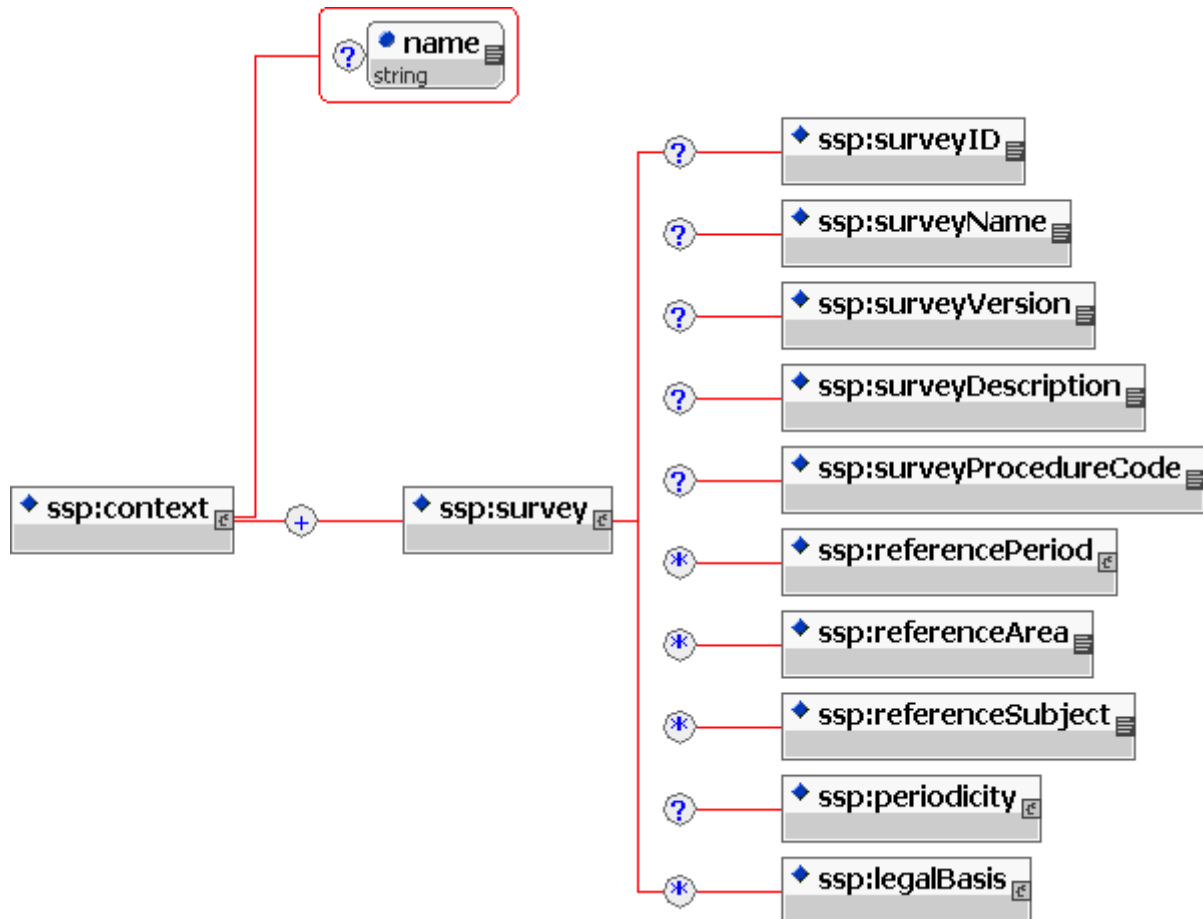
Details

Beschreibt den anwendungsbezogenen Kontext eines Objektes.

8.3. Kontext

8.3.1. Element <ssp:context>

Element	<ssp:context>		
Datentyp:	element-content		
Namespace:	http://www.destatis.de/schema/datml-ssp/1.0/de	DTD:	-



Attribute:	S	Datentyp, Default, Werte
name	?	char:Name des Kontextes

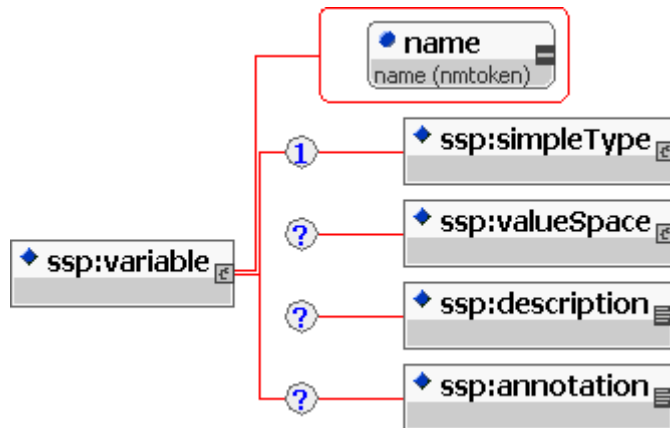
Details

Informationen über den fachlichen, also statistischen Verwendungskontext wie Erhebung, Berichtszeitraum usw. werden im Element <ssp:context> abgelegt

8.4. Daten

8.4.1. Element `<ssp:variable>`

Element	<code><ssp:variable></code>	
Datentyp:	<i>element-content</i>	
Namespace:	<code>http://www.destatis.de/schema/datml-ssp/1.0/de</code>	DTD: -



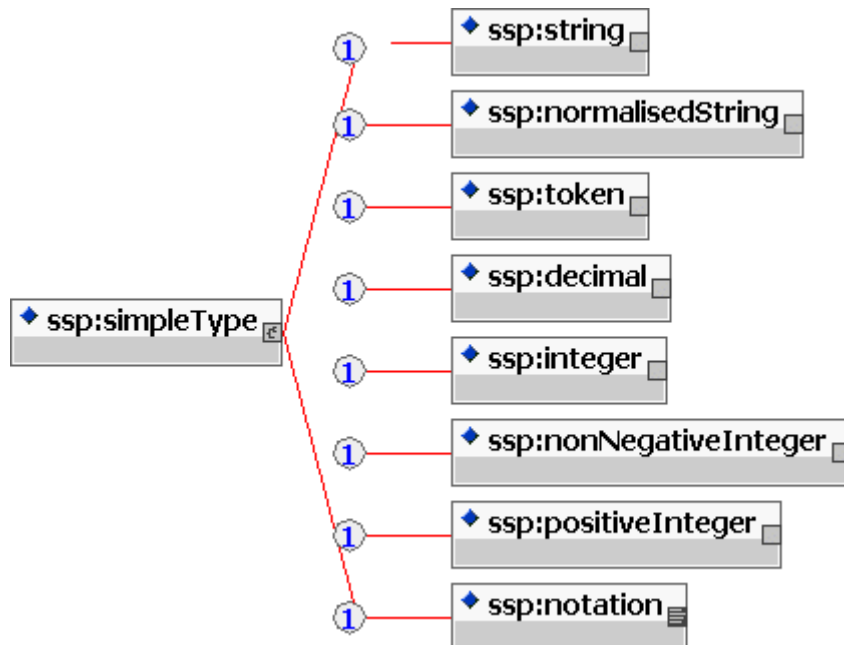
Attribute:	S Datentyp, Default, Werte
name	M <i>char</i> :Name des Merkmals

Details

Beschreibung eines Merkmals

8.4.2. Element `<ssp:simpleType>`

Element	<code><ssp:simpleType></code>	
Datentyp:	<code>element-content</code>	
Namespace:	<code>http://www.destatis.de/schema/datml-ssp/1.0/de</code>	DTD: -



Details

Typbeschreibung eines Merkmals

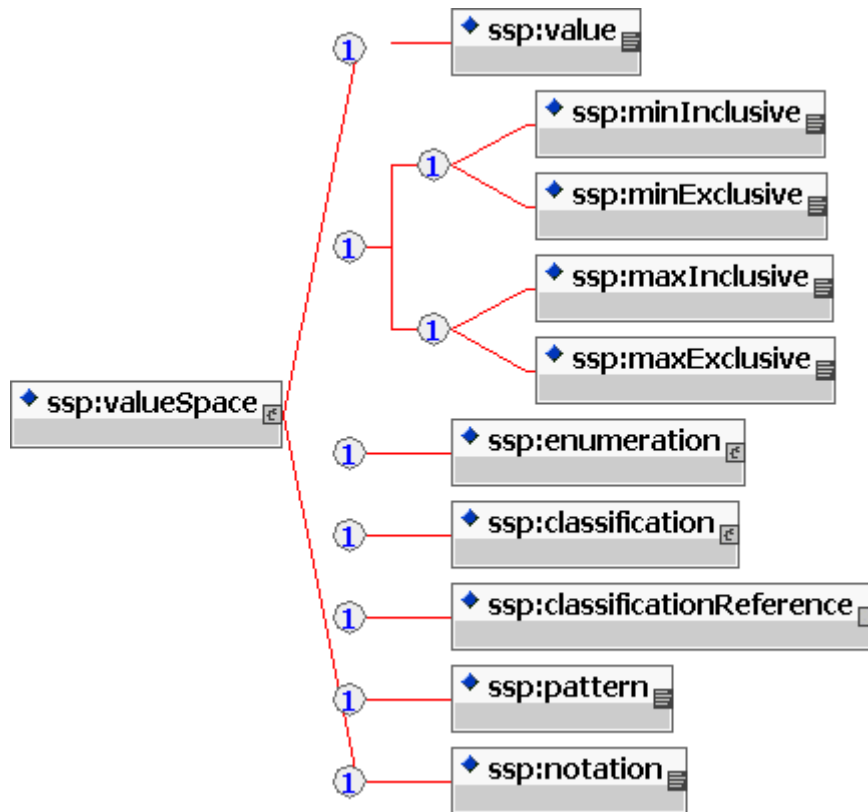
8.4.3. Element <ssp:valueSpace>

Element <ssp:valueSpace>

Datentyp: *element-content*

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

DTD: -

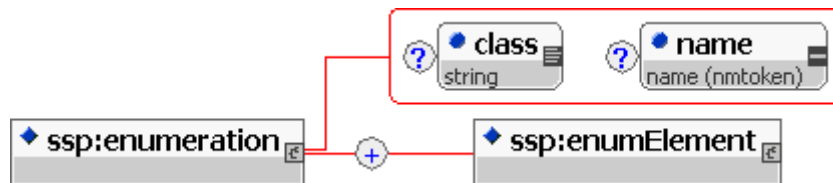


Details

Wertangabe für ein Merkmal

8.4.4. Element <ssp:enumeration>

Element	<ssp:enumeration>		
Datentyp:	<i>element-content</i>		
Namespace:	http://www.destatis.de/schema/datml-ssp/1.0/de	DTD:	-



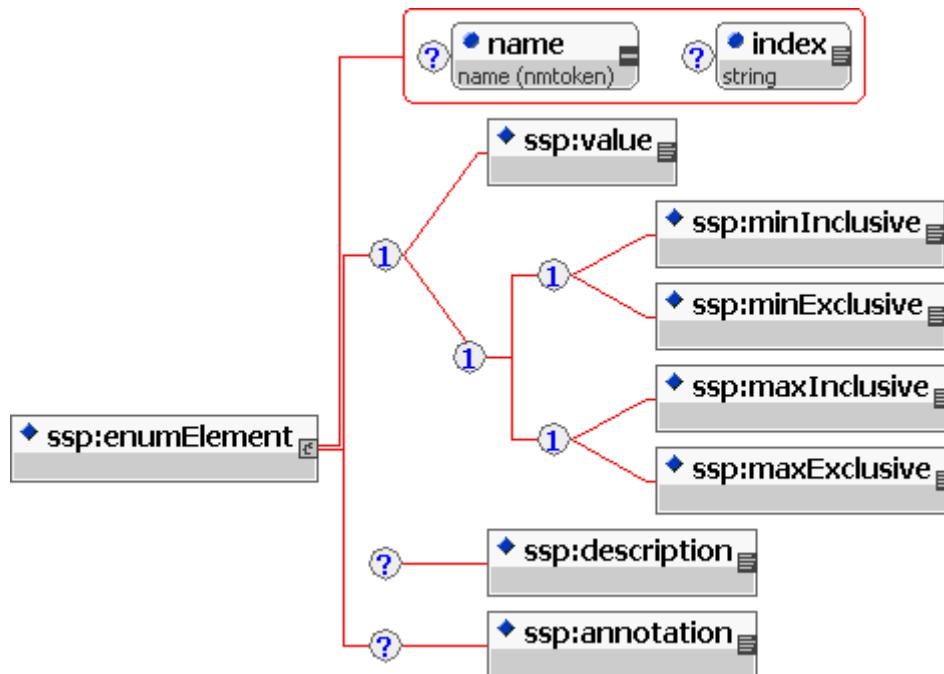
Attribute:	S	Datentyp, Default, Werte
class	?	char:Klassenangabe
name	?	char:Name der Auflistung / derAusprägungsgruppe

Details

Beschreibung einer Ausprägungsgruppe

8.4.5. Element `<ssp:enumElement>`

Element	<code><ssp:enumElement></code>		
Datentyp:	<code>element-content</code>		
Namespace:	<code>http://www.destatis.de/schema/datml-ssp/1.0/de</code>	DTD:	-



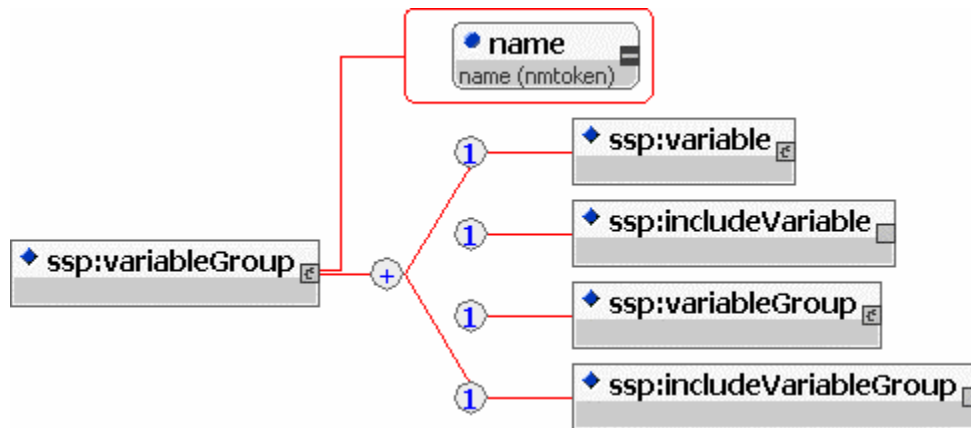
Attribute:	S	Datentyp, <u>D</u> efault, W	erte
name	?	<code>char</code> :Name des Auflistungselementes / der Ausprägung	
index	?	<code>char</code> :Nummer des Auflistungselementes / der Ausprägung	

Details

Beschreibung einer Ausprägung

8.4.6. Element <ssp:variableGroup>

Element	<ssp:variableGroup>		
Datentyp:	<i>element-content</i>		
Namespace:	http://www.destatis.de/schema/datml-ssp/1.0/de	DTD:	-



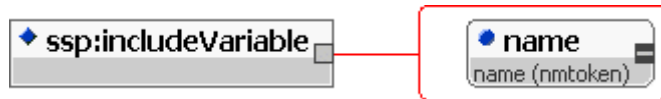
Attribute:	S	Datentyp, <u>Default</u> , Werte
name	M	char:Name der Merkmalsgruppe

Details

Beschreibung einer Merkmalsgruppe, die aus einer Liste von Merkmalen und/oder wiederum Merkmalsgruppen sowie Referenzen auf Merkmale bzw. Merkmalsgruppen besteht.

8.4.7. Element `<ssp:includeVariable>`

Element	<code><ssp:includeVariable></code>		
Datentyp:	<i>empty-element</i>		
Namespace:	<code>http://www.destatis.de/schema/datml-ssp/1.0/de</code>	DTD:	-



Attribute:	S	Datentyp, <u>Default</u>, Werte
<code>name</code>	M	<i>char</i> : Name der zu inkludierenden Variablen

Details

Mit dem Element `<ssp:includeVariable>` wird eine Referenz auf ein Merkmal angegeben.

8.4.8. Element `<ssp:includeVariableGroup>`

Element `<ssp:includeVariableGroup>`

Datentyp: *empty-element*

Namespace: `http://www.destatis.de/schema/datml-ssp/1.0/de` DTD: -



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

`name` **M** *char*: Name der zu inkludierenden Variablengruppe

Details

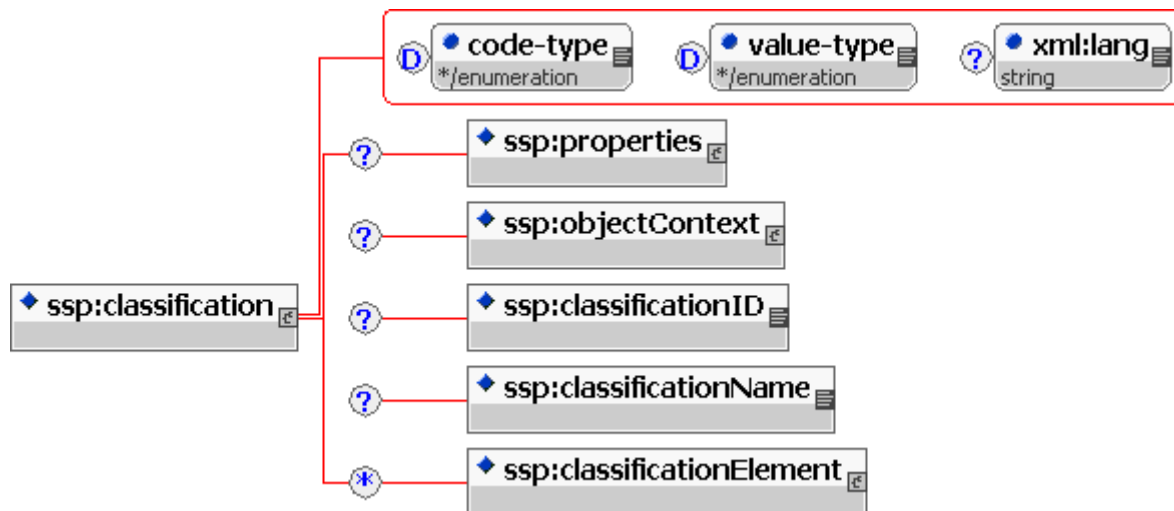
Mit dem Element `<ssp:includeVariableGroup>` wird eine Referenz auf eine Merkmalsgruppe angegeben.

8.5. Klassifikation

Diese Angabe ist optional in einer Erhebungsbeschreibung.

8.5.1. Element <ssp:classification>

Element	<ssp:classification>		
Datentyp:	<i>element-content</i>		
Namespace:	http://www.destatis.de/schema/datml-ssp/1.0/de	DTD:	-



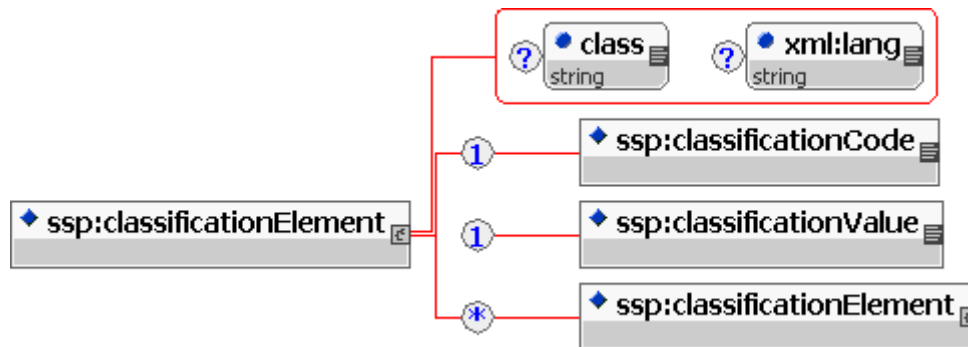
Attribute:	S	Datentyp, <u>Default</u> , Werte
code-type	D	<i>enumeration</i> : (<u>integer</u> string date time); Datentyp des Klassifikationscodes
value-type	D	<i>enumeration</i> : (<u>integer</u> string date time); Datentyp der Klassifikationswerte
xml:lang	?	<i>char</i> ; Bezeichnung einer natürlichen Sprache

Details

Klassifikation

8.5.2. Element `<ssp:classificationElement>`

Element	<code><ssp:classificationElement></code>		
Datentyp:	<code>element-content</code>		
Namespace:	<code>http://www.destatis.de/schema/datml-ssp/1.0/de</code>	DTD:	-



Attribute:	S	Datentyp, Default, Werte
<code>class</code>	?	<code>char</code> : Klasse des Auflistungselementes
<code>xml:lang</code>	?	<code>char</code> ; Bezeichnung einer natürlichen Sprache

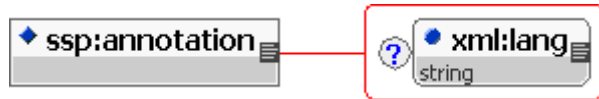
Details

Klassifikationselement

8.6. Basiselementtypen

8.6.1. Element `<ssp:annotation>`

Element	<code><ssp:annotation></code>		
Datentyp:	<i>char</i>		
Namespace:	<code>http://www.destatis.de/schema/datml-ssp/1.0/de</code>	DTD:	-



Attribute:	S	Datentyp, <u>Default</u>, Werte
<code>xml:lang</code>	?	<i>char</i> ; Bezeichnung einer natürlichen Sprache

Details

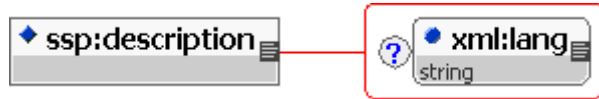
Anmerkungstext

8.6.2. Element `<ssp:description>`

Element `<ssp:description>`

Datentyp: *char*

Namespace: `http://www.destatis.de/schema/datml-ssp/1.0/de` DTD: -



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

`xml:lang` ? *char*; Bezeichnung einer natürlichen Sprache

Details

Beschreibungstext

9. Beispieldokument *[nicht-normativ]*

9.1. Beispiel 1

```

<?xml version = "1.0" encoding="ISO-8859-1"?>
<!--DOCTYPE sdf:surveyDefinition SYSTEM "datml-sdf-1.dtd"-->

<!-- *****
Dies ist ein Dokument zu Testzwecken! Es wird davon ab-
geraten, dieses Dokument in einer Produktivumgebung ein-
zusetzen!
*****
-->
<sdf:surveyDefinition version="1.0"
  xmlns:sdf="http://www.destatis.de/schema/datml-sdf/1.0"
  xmlns:ssp="http://www.destatis.de/schema/datml-ssp/1.0"
  >

  <ssp:properties>
    <ssp:abstract>
      Elektronische Erhebungsbeschreibung (DatML/SDF) für die Bedarfsorien-
      tierte Grundsicherung (ASH) 2004
    </ssp:abstract>
    <ssp:description>
      Diese elektronische Erhebungsbeschreibung im XML-basierten Format
      DatML/SDF beschreibt
      auf standardisierte Weise die Merkmale und Strukturen der Erhebung, so
      daß geeignete
      generische Anwendungen Lieferdokumente im Format DatML/RAW erzeugen
      können.
    </ssp:description>
    <ssp:copyright>Statistisches Bundesamt (Destatis), 2004</ssp:copyright>
    <ssp:author>
      <ssp:fullName>Michael Schäfer</ssp:fullName>
      <ssp:organisation>Statistisches Bundesamt</ssp:organisation>
    </ssp:author>
    <ssp:contact>
      <ssp:person>
        <ssp:fullName>Michael Schäfer</ssp:fullName>
      </ssp:person>
      <ssp:organisation>Statistisches Bundesamt (Destatis)</ssp:organisation>
      <ssp:address>
        <ssp:street>Gustav-Stresemann-Ring</ssp:street>
        <ssp:streetNumber>11 - 13</ssp:streetNumber>
        <ssp:postcode>65189</ssp:postcode>
        <ssp:city>Wiesbaden</ssp:city>
        <ssp:country>Deutschland</ssp:country>
      </ssp:address>
      <ssp:email>michael.schaefer@destatis.de</ssp:email>
      <ssp:fax>0611 75 3951</ssp:fax>
      <ssp:phone>0611 75 3652</ssp:phone>
      <ssp:url caption = "Statistisches Bundesamt (Destatis)">www.destatis.de</ssp:url>
    </ssp:contact>
    <ssp:creationDate>
      <ssp:year>2004</ssp:year>
      <ssp:month>6</ssp:month>
      <ssp:day>02</ssp:day>
    </ssp:creationDate>
    <ssp:creationTime>
      <ssp:hours>10</ssp:hours>
      <ssp:minutes>22</ssp:minutes>
      <ssp:seconds>26</ssp:seconds>
      <ssp:timeZoneName>MEZ</ssp:timeZoneName>
      <ssp:timeZoneOffset utc-vector = "ahead">
        <ssp:hours>1</ssp:hours>
        <ssp:minutes>0</ssp:minutes>
      </ssp:timeZoneOffset>
    </ssp:creationTime>
    <ssp:testIndicator/>
    <ssp:application>
      <ssp:name>UltraEdit32</ssp:name>
      <ssp:version>7.20</ssp:version>
      <ssp:environment>WinXP</ssp:environment>
    </ssp:application>
  </ssp:properties>

```

```

    <ssp:context name = "string">
      <ssp:survey>
        <ssp:surveyID class="ERHID">22141.1.20030101.1</ssp:surveyID>
        <ssp:surveyName>Ausgaben der Sozialhilfe für einmalige Leistungen an
Empfänger, die gleichzeitig
                                Leistungen der Bedarfsorientierten Grundsicherung er-
halten
                                </ssp:surveyName>
        <ssp:surveyVersion>1</ssp:surveyVersion>
        <ssp:referencePeriod>
          <ssp:dateTime>
            <ssp:year>2003</ssp:year>
          </ssp:dateTime>
        </ssp:referencePeriod>
        <ssp:referenceArea>Bundesgebiet</ssp:referenceArea>
        <ssp:referenceSubject>Unternehmen</ssp:referenceSubject>
        <ssp:periodicity>
          <ssp:years>1</ssp:years>
        </ssp:periodicity>
      </ssp:survey>
    </ssp:context>

  <sdf:dataModel>
    <sdf:variables>

      <!-- ErhStelleGemeinde -->

      <ssp:variable name = "ErhStelleGemeinde">
        <ssp:simpleType>
          <ssp:integer total-digits="8"/>
        </ssp:simpleType>
        <ssp:description>
          Gemeindegenschaft des Grundsicherungsamtes
        </ssp:description>
      </ssp:variable>

      <!-- ArtTraeger -->

      <ssp:variable name = "ArtTraeger">
        <ssp:simpleType>
          <ssp:integer total-digits = "1"/>
        </ssp:simpleType>
        <ssp:valueSpace>
          <ssp:enumeration>
            <ssp:enumElement>
              <ssp:value>1</ssp:value>
              <ssp:description>örtlich</ssp:description>
            </ssp:enumElement>
            <ssp:enumElement>
              <ssp:value>2</ssp:value>
              <ssp:description>überörtlich</ssp:description>
            </ssp:enumElement>
          </ssp:enumeration>
        </ssp:valueSpace>
        <ssp:description>
          Die Art des Trägers
        </ssp:description>
      </ssp:variable>

      <!-- Leist18bis64nachPar21 -->

      <ssp:variable name = "Leist18bis64nachPar21">
        <ssp:simpleType>
          <ssp:integer total-digits="10"/>
        </ssp:simpleType>
        <ssp:description>
          Leistungen an Empfänger von 18 bis unter 65 Jahren nach
Paragraph 21 (volle EURO)
        </ssp:description>
      </ssp:variable>

      <!-- Leist18bis64nachPar27 -->

      <ssp:variable name = "Leist18bis64nachPar27">
        <ssp:simpleType>
          <ssp:integer total-digits="10"/>

```

```

        </ssp:simpleType>
        <ssp:description>
            Leistungen an Empfänger von 18 bis unter 65 Jahren nach
Paragraph 27 (volle EURO)
        </ssp:description>
    </ssp:variable>

    <!-- LeistUeber64nachPar21 -->

    <ssp:variable name = "LeistUeber64nachPar21">
        <ssp:simpleType>
            <ssp:integer total-digits="10"/>
        </ssp:simpleType>
        <ssp:description>
            Leistungen an Empfänger von 65 Jahren und älter nach Pa-
Paragraph 21 (volle EURO)
        </ssp:description>
    </ssp:variable>

    <!-- LeistUeber64nachPar27 -->

    <ssp:variable name = "LeistUeber64nachPar27">
        <ssp:simpleType>
            <ssp:integer total-digits="10"/>
        </ssp:simpleType>
        <ssp:description>
            Leistungen an Empfänger von 65 Jahren und älter nach Pa-
Paragraph 27 (volle EURO)
        </ssp:description>
    </ssp:variable>

    </sdf:variables>

</sdf:dataModel>

<sdf:outputDataModel>

    <sdf:group attachment-level="data" required="yes">
        <sdf:includeVariable variable = "ArtTraeger" required = "yes"/>
        <sdf:includeVariable variable = "Leist18bis64nachPar21" required
= "yes"/>
        <sdf:includeVariable variable = "Leist18bis64nachPar27" required
= "yes"/>
        <sdf:includeVariable variable = "LeistUeber64nachPar21" required
= "yes"/>
        <sdf:includeVariable variable = "LeistUeber64nachPar27" required
= "yes"/>
    </sdf:group>

    <sdf:group attachment-level="message" required="yes">
        <sdf:includeVariable variable = "ErhStelleGemeinde" required = "yes"/>
    </sdf:group>

</sdf:outputDataModel>

</sdf:surveyDefinition>

```

9.2. Beispiel 2

```

<?xml version = "1.0" encoding="ISO-8859-1"?>
<!--DOCTYPE sdf:surveyDefinition SYSTEM "datml-sdf-1.dtd"-->

<!-- *****
Dies ist ein Dokument zu Testzwecken! Es wird davon ab-
geraten, dieses Dokument in einer Produktivumgebung ein-
zusetzen!
*****
-->
<sdf:surveyDefinition version="1.0"
  xmlns:sdf="http://www.destatis.de/schema/datml-sdf/1.0"
  xmlns:ssp="http://www.destatis.de/schema/datml-ssp/1.0"
  >

  <ssp:properties>
    <ssp:abstract>
      Elektronische Erhebungsbeschreibung (DatML/SDF) für die Bedarfsorien-
      tierte Grundsicherung (AEG) 2003
    </ssp:abstract>
    <ssp:description>
      Diese elektronische Erhebungsbeschreibung im XML-basierten Format
      DatML/SDF beschreibt
      auf standardisierte Weise die Merkmale und Strukturen der Erhebung, so
      daß geeignete
      generische Anwendungen Lieferdokumente im Format DatML/RAW erzeugen
      können.
    </ssp:description>
    <ssp:copyright>Statistisches Bundesamt (Destatis), 2004</ssp:copyright>
    <ssp:author>
      <ssp:fullName>Michael Schäfer</ssp:fullName>
      <ssp:organisation>Statistisches Bundesamt</ssp:organisation>
    </ssp:author>
    <ssp:contact>
      <ssp:person>
        <ssp:fullName>Michael Schäfer</ssp:fullName>
      </ssp:person>
      <ssp:organisation>Statistisches Bundesamt (Destatis)</ssp:organisation>
      <ssp:address>
        <ssp:street>Gustav-Stresemann-Ring</ssp:street>
        <ssp:streetNumber>11 - 13</ssp:streetNumber>
        <ssp:postcode>65189</ssp:postcode>
        <ssp:city>Wiesbaden</ssp:city>
        <ssp:country>Deutschland</ssp:country>
      </ssp:address>
      <ssp:email>michael.schaefer@destatis.de</ssp:email>
      <ssp:fax>0611 75 3951</ssp:fax>
      <ssp:phone>0611 75 3652</ssp:phone>
      <ssp:url caption = "Statistisches Bundesamt (Destatis)">www.destatis.de</ssp:url>
    </ssp:contact>
    <ssp:creationDate>
      <ssp:year>2004</ssp:year>
      <ssp:month>6</ssp:month>
      <ssp:day>02</ssp:day>
    </ssp:creationDate>
    <ssp:creationTime>
      <ssp:hours>18</ssp:hours>
      <ssp:minutes>56</ssp:minutes>
      <ssp:seconds>26</ssp:seconds>
      <ssp:timeZoneName>MEZ</ssp:timeZoneName>
      <ssp:timeZoneOffset utc-vector = "ahead">
        <ssp:hours>1</ssp:hours>
        <ssp:minutes>0</ssp:minutes>
      </ssp:timeZoneOffset>
    </ssp:creationTime>
    <ssp:testIndicator/>
    <ssp:application>
      <ssp:name>UltraEdit32</ssp:name>
      <ssp:version>7.20</ssp:version>
      <ssp:environment>WinXP</ssp:environment>
    </ssp:application>
  </ssp:properties>

  <ssp:context>
    <ssp:survey>

```

```

    <ssp:surveyID class="ERHID">22951.1.20030101.1</ssp:surveyID>
    <ssp:surveyName>Ausgabe und Einnahmen für die Bedarfsorientierte Grund-
sicherung</ssp:surveyName>
    <ssp:surveyVersion>1</ssp:surveyVersion>
    <ssp:referencePeriod>
      <ssp:dateTime>
        <ssp:year>2003</ssp:year>
      </ssp:dateTime>
    </ssp:referencePeriod>
    <ssp:referenceArea>Bundesgebiet</ssp:referenceArea>
    <ssp:referenceSubject>Grundsicherungsämter</ssp:referenceSubject>
    <ssp:periodicity>
      <ssp:years>1</ssp:years>
    </ssp:periodicity>
  </ssp:survey>
</ssp:context>

<sdf:dataModel>
  <sdf:variables>

    <!-- ErhStelleGemeinde -->

    <ssp:variable name = "ErhStelleGemeinde">
      <ssp:simpleType>
        <ssp:integer total-digits="8"/>
      </ssp:simpleType>
      <ssp:description>
        Gemeindegennzahl des Grundsicherungsamtes
      </ssp:description>
    </ssp:variable>

    <!-- AusgabenAusEinrichtungen -->

    <ssp:variable name = "AusgabenAusEinrichtungen">
      <ssp:simpleType>
        <ssp:integer total-digits="10"/>
      </ssp:simpleType>
      <ssp:description>
        Ausgaben für Leistungen außerhalb von Einrichtungen (vol-
le Euro)
      </ssp:description>
    </ssp:variable>

    <!-- AusgabenInnEinrichtungen -->

    <ssp:variable name = "AusgabenInnEinrichtungen">
      <ssp:simpleType>
        <ssp:integer total-digits="10"/>
      </ssp:simpleType>
      <ssp:description>
        Ausgaben für Leistungen innerhalb von Einrichtungen (vol-
le Euro)
      </ssp:description>
    </ssp:variable>

    <!-- AusgabenInsgesamt -->

    <ssp:variable name = "AusgabenInsgesamt">
      <ssp:simpleType>
        <ssp:integer total-digits="10"/>
      </ssp:simpleType>
      <ssp:description>
        Ausgaben für Leistungen insgesamt (volle Euro)
      </ssp:description>
    </ssp:variable>

    <!-- EinnahmenAusEinrichtungen -->

    <ssp:variable name = "EinnahmenAusEinrichtungen">
      <ssp:simpleType>
        <ssp:integer total-digits="10"/>
      </ssp:simpleType>
      <ssp:description>
        Einnahmen außerhalb von Einrichtungen (volle Euro)
      </ssp:description>
    </ssp:variable>

    <!-- EinnahmenInnEinrichtungen -->

```

```

    <ssp:variable name = "EinnahmenInnEinrichtungen">
      <ssp:simpleType>
        <ssp:integer total-digits="10"/>
      </ssp:simpleType>
      <ssp:description>
        Einnahmen innerhalb von Einrichtungen (volle Euro)
      </ssp:description>
    </ssp:variable>

    <!-- EinnahmenInsgesamt -->

    <ssp:variable name = "EinnahmenInsgesamt">
      <ssp:simpleType>
        <ssp:integer total-digits="10"/>
      </ssp:simpleType>
      <ssp:description>
        Einnahmen insgesamt (volle Euro)
      </ssp:description>
    </ssp:variable>

    <!-- KostenGutachten -->

    <ssp:variable name = "KostenGutachten">
      <ssp:simpleType>
        <ssp:integer total-digits="8"/>
      </ssp:simpleType>
      <ssp:description>
        Kosten für abgeschlossene Gutachten (volle Euro)
      </ssp:description>
    </ssp:variable>

    <!-- AnzahlGutachten -->

    <ssp:variable name = "AnzahlGutachten">
      <ssp:simpleType>
        <ssp:integer total-digits="5"/>
      </ssp:simpleType>
      <ssp:description>
        Anzahl der abgeschlossenen Gutachten
      </ssp:description>
    </ssp:variable>

  </sdf:variables>

</sdf:dataModel>

<sdf:outputDataModel>

  <sdf:group attachment-level="data" required = "yes">
    <sdf:includeVariable variable = "AusgabenAusEinrichtungen" required =
"yes"/>
    <sdf:includeVariable variable = "AusgabenInnEinrichtungen" required =
"yes"/>
    <sdf:includeVariable variable = "AusgabenInsgesamt" required = "yes"/>
    <sdf:includeVariable variable = "EinnahmenAusEinrichtungen" required =
"yes"/>
    <sdf:includeVariable variable = "EinnahmenInnEinrichtungen" required =
"yes"/>
    <sdf:includeVariable variable = "EinnahmenInsgesamt" required = "yes"/>
    <sdf:includeVariable variable = "KostenGutachten" required = "yes"/>
    <sdf:includeVariable variable = "AnzahlGutachten" required = "yes"/>
  </sdf:group>

  <sdf:group attachment-level="message" required="yes">
    <sdf:includeVariable variable = "ErhStelleGemeinde" required = "yes"/>
  </sdf:group>

</sdf:outputDataModel>

</sdf:surveyDefinition>

```

9.3. Beispiel 3

```

<?xml version = "1.0" encoding="ISO-8859-1"?>
<!--DOCTYPE sdf:surveyDefinition SYSTEM "datml-sdf-1.dtd"-->

<!-- *****
Dies ist ein Dokument zu Testzwecken! Es wird davon ab-
geraten, dieses Dokument in einer Produktivumgebung ein-
zusetzen!
*****
-->
<sdf:surveyDefinition version="1.0"
  xmlns:sdf="http://www.destatis.de/schema/datml/sdf/1.0"
  xmlns:ssp="http://www.destatis.de/schema/DatML/SSP/1.0"
  >

  <ssp:properties>
    <ssp:abstract>
      Elektronische Erhebungsbeschreibung (DatML/SDF) für die Bedarfsorien-
      tierte Grundsicherung (EBG) 2003
    </ssp:abstract>
    <ssp:description>
      Diese elektronische Erhebungsbeschreibung im XML-basierten Format
      DatML/SDF beschreibt
      auf standardisierte Weise die Merkmale und Strukturen der Erhebung, so
      daß geeignete
      generische Anwendungen Lieferdokumente im Format DatML/RAW erzeugen
      können.
    </ssp:description>
    <ssp:copyright>Statistisches Bundesamt (Destatis), 2004</ssp:copyright>
    <ssp:author>
      <ssp:fullName>Michael Schäfer</ssp:fullName>
      <ssp:organisation>Statistisches Bundesamt</ssp:organisation>
    </ssp:author>
    <ssp:contact>
      <ssp:person>
        <ssp:fullName>Michael Schäfer</ssp:fullName>
      </ssp:person>
      <ssp:organisation>Statistisches Bundesamt (Destatis)</ssp:organisation>
      <ssp:address>
        <ssp:street>Gustav-Stresemann-Ring</ssp:street>
        <ssp:streetNumber>11 - 13</ssp:streetNumber>
        <ssp:postcode>65189</ssp:postcode>
        <ssp:city>Wiesbaden</ssp:city>
        <ssp:country>Deutschland</ssp:country>
      </ssp:address>
      <ssp:email>michael.schaefer@destatis.de</ssp:email>
      <ssp:fax>0611 75 3951</ssp:fax>
      <ssp:phone>0611 75 3652</ssp:phone>
      <ssp:url caption = "Statistisches Bundesamt (Destatis)">www.destatis.de</ssp:url>
    </ssp:contact>
    <ssp:creationDate>
      <ssp:year>2004</ssp:year>
      <ssp:month>6</ssp:month>
      <ssp:day>02</ssp:day>
    </ssp:creationDate>
    <ssp:creationTime>
      <ssp:hours>20</ssp:hours>
      <ssp:minutes>19</ssp:minutes>
      <ssp:seconds>26</ssp:seconds>
      <ssp:timeZoneName>MEZ</ssp:timeZoneName>
      <ssp:timeZoneOffset utc-vector = "ahead">
        <ssp:hours>1</ssp:hours>
        <ssp:minutes>0</ssp:minutes>
      </ssp:timeZoneOffset>
    </ssp:creationTime>
    <ssp:testIndicator/>
    <ssp:application>
      <ssp:name>UltraEdit32</ssp:name>
      <ssp:version>7.20</ssp:version>
      <ssp:environment>WinXP</ssp:environment>
    </ssp:application>
  </ssp:properties>

  <ssp:context>
    <ssp:survey>

```

```

    <ssp:surveyID class="ERHID">22961.1.20030101.1</ssp:surveyID>
    <ssp:surveyName>Empfänger von Bedarfsorientierter Grundsiche-
rungs</ssp:surveyName>
    <ssp:surveyVersion>1</ssp:surveyVersion>
    <ssp:referencePeriod>
      <ssp:dateTime>
        <ssp:year>2003</ssp:year>
      </ssp:dateTime>
    </ssp:referencePeriod>
    <ssp:referenceArea>Bundesgebiet</ssp:referenceArea>
    <ssp:referenceSubject>Grundsicherungsämter</ssp:referenceSubject>
    <ssp:periodicity>
      <ssp:years>1</ssp:years>
    </ssp:periodicity>
  </ssp:survey>
</ssp:context>

<sdf:dataModel>
  <sdf:variables>

    <!-- ErhStelleGemeinde -->

    <ssp:variable name = "ErhStelleGemeinde">
      <ssp:simpleType>
        <ssp:integer total-digits="8"/>
      </ssp:simpleType>
      <ssp:description>
        Gemeindegenschaft des Grundsicherungsamtes
      </ssp:description>
    </ssp:variable>

    <!-- KennNrInGemeinde -->

    <ssp:variable name = "KennNrInGemeinde">
      <ssp:simpleType>
        <ssp:integer total-digits="8"/>
      </ssp:simpleType>
      <ssp:description>
        Kennnummer des Grundsicherungsamtes innerhalb der Gemeinde
      </ssp:description>
    </ssp:variable>

    <!-- WohnortEmpfGemeinde -->

    <ssp:variable name = "WohnortEmpfGemeinde">
      <ssp:simpleType>
        <ssp:integer total-digits="8"/>
      </ssp:simpleType>
      <ssp:description>
        Gemeindegenschaft des Wohnortes des Empfängers
      </ssp:description>
    </ssp:variable>

    <!-- WohnortEmpfGemTeil -->

    <ssp:variable name = "WohnortEmpfGemTeil">
      <ssp:simpleType>
        <ssp:integer total-digits="8"/>
      </ssp:simpleType>
      <ssp:description>
        Kennnummer des Wohnortes des Empfängers innerhalb der Ge-
meinde
      </ssp:description>
    </ssp:variable>

    <!-- PersNr -->

    <ssp:variable name = "PersNr">
      <ssp:simpleType>
        <ssp:integer total-digits="1"/>
      </ssp:simpleType>
      <ssp:valueSpace>
        <ssp:enumeration>
          <ssp:enumElement name="Person1">
            <ssp:value>1</ssp:value>
          </ssp:enumElement>
          <ssp:enumElement name="Person2">
            <ssp:value>2</ssp:value>
          </ssp:enumElement>
        </ssp:enumeration>
      </ssp:valueSpace>
    </ssp:variable>
  </sdf:variables>
</sdf:dataModel>

```

```

        </ssp:enumElement>
    </ssp:enumeration>
</ssp:valueSpace>
<ssp:description>
    Nummer der Person (1 oder 2)
</ssp:description>
</ssp:variable>

<!-- Geschlecht -->

<ssp:variable name = "Geschlecht">
    <ssp:simpleType>
        <ssp:integer total-digits="1"/>
    </ssp:simpleType>
    <ssp:valueSpace>
        <ssp:enumeration>
            <ssp:enumElement name="Männlich">
                <ssp:value>1</ssp:value>
            </ssp:enumElement>
            <ssp:enumElement name="Weiblich">
                <ssp:value>2</ssp:value>
            </ssp:enumElement>
        </ssp:enumeration>
    </ssp:valueSpace>
    <ssp:description>
        Geschlecht; 1=männlich, 2=weiblich
    </ssp:description>
</ssp:variable>

<!-- Geburtsmonat -->

<ssp:variable name = "Geburtsmonat">
    <ssp:simpleType>
        <ssp:integer total-digits="2"/>
    </ssp:simpleType>
    <ssp:valueSpace>
        <ssp:minValue>1</ssp:minValue>
        <ssp:maxValue>12</ssp:maxValue>
    </ssp:valueSpace>
    <ssp:description>
        Der Geburtsmonat der Person
    </ssp:description>
</ssp:variable>

<!-- Geburtsjahr -->

<ssp:variable name = "Geburtsjahr">
    <ssp:simpleType>
        <ssp:integer total-digits="4"/>
    </ssp:simpleType>
    <ssp:description>
        Das Geburtsjahr der Person
    </ssp:description>
</ssp:variable>

<!-- StaatsAng -->

<ssp:variable name = "StaatsAng">
    <ssp:simpleType>
        <ssp:integer total-digits="1"/>
    </ssp:simpleType>
    <ssp:valueSpace>
        <ssp:enumeration>
            <ssp:enumElement name="Deutsch">
                <ssp:value>1</ssp:value>
            </ssp:enumElement>
            <ssp:enumElement name="EU-Ausländer">
                <ssp:value>2</ssp:value>
            </ssp:enumElement>
            <ssp:enumElement name="Sonstige Ausländer">
                <ssp:value>3</ssp:value>
            </ssp:enumElement>
        </ssp:enumeration>
    </ssp:valueSpace>
    <ssp:description>
        Staatsangehörigkeit
    </ssp:description>
</ssp:variable>

```

```

<!-- VolleErwerbsMind -->
<ssp:variable name = "VolleErwerbsMind">
  <ssp:simpleType>
    <ssp:integer total-digits="1"/>
  </ssp:simpleType>
  <ssp:valueSpace>
    <ssp:enumeration>
      <ssp:enumElement name="Ja">
        <ssp:value>1</ssp:value>
      </ssp:enumElement>
    </ssp:enumeration>
  </ssp:valueSpace>
  <ssp:description>
    Volle Erwerbsminderung gem. §1 Nr.2 GSIG; ja=1, nein=leer
  </ssp:description>
</ssp:variable>

<!-- GSGewaeht -->
<ssp:variable name = "GSGewaeht">
  <ssp:simpleType>
    <ssp:integer total-digits="1"/>
  </ssp:simpleType>
  <ssp:valueSpace>
    <ssp:enumeration>
      <ssp:enumElement na-
me="GewaehtAusserhalbEinrichtung">
        <ssp:value>1</ssp:value>
      </ssp:enumElement>
      <ssp:enumElement na-
me="GewaehtInnerhalbEinrichtung">
        <ssp:value>2</ssp:value>
      </ssp:enumElement>
    </ssp:enumeration>
  </ssp:valueSpace>
  <ssp:description>
    Grundsicherung wurde innerhalb (=1) bzw. auerhalb (=2)
von Einrichtungen gewaeht
  </ssp:description>
</ssp:variable>

<!-- LeistBeginnMonat -->
<ssp:variable name = "LeistBeginnMonat">
  <ssp:simpleType>
    <ssp:integer total-digits="2"/>
  </ssp:simpleType>
  <ssp:valueSpace>
    <ssp:minValue>1</ssp:minValue>
    <ssp:maxValue>12</ssp:maxValue>
  </ssp:valueSpace>
  <ssp:description>
    Der Monat, in dem die Leistung beginnt
  </ssp:description>
</ssp:variable>

<!-- LeistBeginnJahr -->
<ssp:variable name = "LeistBeginnJahr">
  <ssp:simpleType>
    <ssp:integer total-digits="4"/>
  </ssp:simpleType>
  <ssp:description>
    Das Jahr, in dem die Leistung beginnt
  </ssp:description>
</ssp:variable>

<!-- GewUrsache1 -->
<ssp:variable name = "GewUrsache1">
  <ssp:simpleType>
    <ssp:integer total-digits="1"/>
  </ssp:simpleType>
  <ssp:valueSpace>
    <ssp:enumeration>
      <ssp:enumElement name="?">

```

```

        <ssp:value>1</ssp:value>
    </ssp:enumElement>
    <ssp:enumElement name=""?>
        <ssp:value>2</ssp:value>
    </ssp:enumElement>
    <ssp:enumElement name=""?>
        <ssp:value>3</ssp:value>
    </ssp:enumElement>
    <ssp:enumElement name=""?>
        <ssp:value>4</ssp:value>
    </ssp:enumElement>
    <ssp:enumElement name=""?>
        <ssp:value>5</ssp:value>
    </ssp:enumElement>
    <ssp:enumElement name=""?>
        <ssp:value>6</ssp:value>
    </ssp:enumElement>
    <ssp:enumElement name=""?>
        <ssp:value>7</ssp:value>
    </ssp:enumElement>
    <ssp:enumElement name=""?>
        <ssp:value>8</ssp:value>
    </ssp:enumElement>
</ssp:enumeration>
</ssp:valueSpace>
<ssp:description>
    1. Ursache der Gewährung der Grundsicherung (1..8 gem.
Schlüsselverzeichnis)
    </ssp:description>
</ssp:variable>

<!-- GewUrsache2 -->

<ssp:variable name = "GewUrsache2">
    <ssp:simpleType>
        <ssp:integer total-digits="1"/>
    </ssp:simpleType>
    <ssp:valueSpace>
        <ssp:enumeration>
            <ssp:enumElement name=""?>
                <ssp:value>1</ssp:value>
            </ssp:enumElement>
            <ssp:enumElement name=""?>
                <ssp:value>2</ssp:value>
            </ssp:enumElement>
            <ssp:enumElement name=""?>
                <ssp:value>3</ssp:value>
            </ssp:enumElement>
            <ssp:enumElement name=""?>
                <ssp:value>4</ssp:value>
            </ssp:enumElement>
            <ssp:enumElement name=""?>
                <ssp:value>5</ssp:value>
            </ssp:enumElement>
            <ssp:enumElement name=""?>
                <ssp:value>6</ssp:value>
            </ssp:enumElement>
            <ssp:enumElement name=""?>
                <ssp:value>7</ssp:value>
            </ssp:enumElement>
            <ssp:enumElement name=""?>
                <ssp:value>8</ssp:value>
            </ssp:enumElement>
        </ssp:enumeration>
    </ssp:valueSpace>
    <ssp:description>
        2. Ursache der Gewährung der Grundsicherung (1..8 gem.
Schlüsselverzeichnis)
    </ssp:description>
</ssp:variable>

<!-- NettoAnspruch -->

<ssp:variable name = "NettoAnspruch">
    <ssp:simpleType>
        <ssp:integer total-digits="4"/>
    </ssp:simpleType>
    <ssp:description>

```

```

        Nettoanspruch (volle Euro)
    </ssp:description>
</ssp:variable>

<!-- Regelanspruch -->

<ssp:variable name = "Regelanspruch">
    <ssp:simpleType>
        <ssp:integer total-digits="4"/>
    </ssp:simpleType>
    <ssp:description>
        Regelanspruch (volle Euro)
    </ssp:description>
</ssp:variable>

<!-- AnerkAufwend -->

<ssp:variable name = "AnerkAufwend">
    <ssp:simpleType>
        <ssp:integer total-digits="4"/>
    </ssp:simpleType>
    <ssp:description>
        Anerkannte Aufwendungen für Unterkunft, Heizung (volle
Euro)
    </ssp:description>
</ssp:variable>

<!-- KrankBeitrag -->

<ssp:variable name = "KrankBeitrag">
    <ssp:simpleType>
        <ssp:integer total-digits="4"/>
    </ssp:simpleType>
    <ssp:description>
        Übernommene Beiträge für die Kranken- und Pflegeversiche-
rung (volle Euro)
    </ssp:description>
</ssp:variable>

<!-- Mehrbedarf -->

<ssp:variable name = "Mehrbedarf">
    <ssp:simpleType>
        <ssp:integer total-digits="4"/>
    </ssp:simpleType>
    <ssp:description>
        Mehrbedarf des Empfängers (volle Euro)
    </ssp:description>
</ssp:variable>

<!-- KeinEink -->

<ssp:variable name = "KeinEink">
    <ssp:simpleType>
        <ssp:integer total-digits="1"/>
    </ssp:simpleType>
    <ssp:valueSpace>
        <ssp:enumeration>
            <ssp:enumElement name="Ja">
                <ssp:value>1</ssp:value>
            </ssp:enumElement>
        </ssp:enumeration>
    </ssp:valueSpace>
    <ssp:description>
        Art des angerechneten/in Anspruch genommenen Einkommens:
Kein Einkommen; ja=1, nein=leer
    </ssp:description>
</ssp:variable>

<!-- ErwerbsEink -->

<ssp:variable name = "ErwerbsEink">
    <ssp:simpleType>
        <ssp:integer total-digits="1"/>
    </ssp:simpleType>
    <ssp:valueSpace>
        <ssp:enumeration>
            <ssp:enumElement name="Ja">

```

```

        <ssp:value>1</ssp:value>
      </ssp:enumElement>
    </ssp:enumeration>
  </ssp:valueSpace>
  <ssp:description>
    Art des angerechneten/in Anspuch genommenen Einkommens:
Erwerbseinkommen; ja=1, nein=leer
  </ssp:description>
</ssp:variable>

<!-- RenteErwerbsMind -->

<ssp:variable name = "RenteErwerbsMind">
  <ssp:simpleType>
    <ssp:integer total-digits="1"/>
  </ssp:simpleType>
  <ssp:valueSpace>
    <ssp:enumeration>
      <ssp:enumElement name="Ja">
        <ssp:value>1</ssp:value>
      </ssp:enumElement>
    </ssp:enumeration>
  </ssp:valueSpace>
  <ssp:description>
    Art des angerechneten/in Anspuch genommenen Einkommens:
Rente wegen Erwerbsminderung; ja=1, nein=leer
  </ssp:description>
</ssp:variable>

<!-- RenteAlter -->

<ssp:variable name = "RenteAlter">
  <ssp:simpleType>
    <ssp:integer total-digits="1"/>
  </ssp:simpleType>
  <ssp:valueSpace>
    <ssp:enumeration>
      <ssp:enumElement name="Ja">
        <ssp:value>1</ssp:value>
      </ssp:enumElement>
    </ssp:enumeration>
  </ssp:valueSpace>
  <ssp:description>
    Art des angerechneten/in Anspuch genommenen Einkommens:
Altersrente; ja=1, nein=leer
  </ssp:description>
</ssp:variable>

<!-- RenteHinterblieb -->

<ssp:variable name = "RenteHinterblieb">
  <ssp:simpleType>
    <ssp:integer total-digits="1"/>
  </ssp:simpleType>
  <ssp:valueSpace>
    <ssp:enumeration>
      <ssp:enumElement name="Ja">
        <ssp:value>1</ssp:value>
      </ssp:enumElement>
    </ssp:enumeration>
  </ssp:valueSpace>
  <ssp:description>
    Art des angerechneten/in Anspuch genommenen Einkommens:
Hinterbliebenenrente; ja=1, nein=leer
  </ssp:description>
</ssp:variable>

<!-- RentePrivAltVorsorge -->

<ssp:variable name = "RentePrivAltVorsorge">
  <ssp:simpleType>
    <ssp:integer total-digits="1"/>
  </ssp:simpleType>
  <ssp:valueSpace>
    <ssp:enumeration>
      <ssp:enumElement name="Ja">
        <ssp:value>1</ssp:value>
      </ssp:enumElement>
    </ssp:enumeration>
  </ssp:valueSpace>
  <ssp:description>
    Art des angerechneten/in Anspuch genommenen Einkommens:

```

```

        </ssp:enumeration>
    </ssp:valueSpace>
    <ssp:description>
        Art des angerechneten/in Anspuch genommenen Einkommens:
Renten aus staatlich geförderter Altersvorsorge; ja=1, nein=leer
    </ssp:description>
</ssp:variable>

<!-- RenteSonstVorsorge -->

<ssp:variable name = "RenteSonstVorsorge">
    <ssp:simpleType>
        <ssp:integer total-digits="1"/>
    </ssp:simpleType>
    <ssp:valueSpace>
        <ssp:enumeration>
            <ssp:enumElement name="Ja">
                <ssp:value>1</ssp:value>
            </ssp:enumElement>
        </ssp:enumeration>
    </ssp:valueSpace>
    <ssp:description>
        Art des angerechneten/in Anspuch genommenen Einkommens:
Renten aus sonstiger privater Vorsorge; ja=1, nein=leer
    </ssp:description>
</ssp:variable>

<!-- VersorgBezuege -->

<ssp:variable name = "VersorgBezuege">
    <ssp:simpleType>
        <ssp:integer total-digits="1"/>
    </ssp:simpleType>
    <ssp:valueSpace>
        <ssp:enumeration>
            <ssp:enumElement name="Ja">
                <ssp:value>1</ssp:value>
            </ssp:enumElement>
        </ssp:enumeration>
    </ssp:valueSpace>
    <ssp:description>
        Art des angerechneten/in Anspuch genommenen Einkommens:
Versorgungsbezüge; ja=1, nein=leer
    </ssp:description>
</ssp:variable>

<!-- RenteBetrAltVorsorge -->

<ssp:variable name = "RenteBetrAltVorsorge">
    <ssp:simpleType>
        <ssp:integer total-digits="1"/>
    </ssp:simpleType>
    <ssp:valueSpace>
        <ssp:enumeration>
            <ssp:enumElement name="Ja">
                <ssp:value>1</ssp:value>
            </ssp:enumElement>
        </ssp:enumeration>
    </ssp:valueSpace>
    <ssp:description>
        Art des angerechneten/in Anspuch genommenen Einkommens:
Renten aus betrieblicher Altersvorsorge; ja=1, nein=leer
    </ssp:description>
</ssp:variable>

<!-- LeistGKV -->

<ssp:variable name = "LeistGKV">
    <ssp:simpleType>
        <ssp:integer total-digits="1"/>
    </ssp:simpleType>
    <ssp:valueSpace>
        <ssp:enumeration>
            <ssp:enumElement name="Ja">
                <ssp:value>1</ssp:value>
            </ssp:enumElement>
        </ssp:enumeration>
    </ssp:valueSpace>

```

```

        <ssp:description>
            Art des angerechneten/in Anspuch genommenen Einkommens:
Leistungen aus der GKV; ja=1, nein=leer
        </ssp:description>
    </ssp:variable>

<!-- Kindergeld -->

<ssp:variable name = "Kindergeld">
    <ssp:simpleType>
        <ssp:integer total-digits="1"/>
    </ssp:simpleType>
    <ssp:valueSpace>
        <ssp:enumeration>
            <ssp:enumElement name="Ja">
                <ssp:value>1</ssp:value>
            </ssp:enumElement>
        </ssp:enumeration>
    </ssp:valueSpace>
    <ssp:description>
        Art des angerechneten/in Anspuch genommenen Einkommens:
Kindergeld; ja=1, nein=leer
    </ssp:description>
</ssp:variable>

<!-- Wohngeld -->

<ssp:variable name = "Wohngeld">
    <ssp:simpleType>
        <ssp:integer total-digits="1"/>
    </ssp:simpleType>
    <ssp:valueSpace>
        <ssp:enumeration>
            <ssp:enumElement name="Ja">
                <ssp:value>1</ssp:value>
            </ssp:enumElement>
        </ssp:enumeration>
    </ssp:valueSpace>
    <ssp:description>
        Art des angerechneten/in Anspuch genommenen Einkommens:
Wohngeld; ja=1, nein=leer
    </ssp:description>
</ssp:variable>

<!-- PrivUnterhalt -->

<ssp:variable name = "PrivUnterhalt">
    <ssp:simpleType>
        <ssp:integer total-digits="1"/>
    </ssp:simpleType>
    <ssp:valueSpace>
        <ssp:enumeration>
            <ssp:enumElement name="Ja">
                <ssp:value>1</ssp:value>
            </ssp:enumElement>
        </ssp:enumeration>
    </ssp:valueSpace>
    <ssp:description>
        Art des angerechneten/in Anspuch genommenen Einkommens:
Private Unterhaltsleistungen; ja=1, nein=leer
    </ssp:description>
</ssp:variable>

<!-- EinkEhe -->

<ssp:variable name = "EinkEhe">
    <ssp:simpleType>
        <ssp:integer total-digits="1"/>
    </ssp:simpleType>
    <ssp:valueSpace>
        <ssp:enumeration>
            <ssp:enumElement name="Ja">
                <ssp:value>1</ssp:value>
            </ssp:enumElement>
        </ssp:enumeration>
    </ssp:valueSpace>
    <ssp:description>

```

```

        Art des angerechneten/in Anspuch genommenen Einkommens:
Einkünfte aus Ehe oder eheähnlicher Gemeinschaft; ja=1, nein=leer
        </ssp:description>
    </ssp:variable>

    <!-- SonstEink -->

    <ssp:variable name = "SonstEink">
        <ssp:simpleType>
            <ssp:integer total-digits="1"/>
        </ssp:simpleType>
        <ssp:valueSpace>
            <ssp:enumeration>
                <ssp:enumElement name="Ja">
                    <ssp:value>1</ssp:value>
                </ssp:enumElement>
            </ssp:enumeration>
        </ssp:valueSpace>
        <ssp:description>
            Art des angerechneten/in Anspuch genommenen Einkommens:
Sonstige Einkünfte; ja=1, nein=leer
        </ssp:description>
    </ssp:variable>

</sdf:variables>

</sdf:dataModel>

<sdf:outputDataModel>

    <sdf:group attachment-level="data" required = "yes">
        <sdf:includeVariable variable = "WohnortEmpfGemeinde" required = "yes"/>
        <sdf:includeVariable variable = "WohnortEmpfGemTeil" required = "condi-
tional"/>
        <sdf:includeVariable variable = "PersNr" required = "yes"/>
        <sdf:includeVariable variable = "Geschlecht" required = "yes"/>
        <sdf:includeVariable variable = "Geburtsmonat" required = "yes"/>
        <sdf:includeVariable variable = "Geburtsjahr" required = "yes"/>
        <sdf:includeVariable variable = "StaatsAng" required = "yes"/>
        <sdf:includeVariable variable = "VolleErwerbsMind" required = "condi-
tional"/>
        <sdf:includeVariable variable = "GSGewaeht" required = "yes"/>
        <sdf:includeVariable variable = "LeistBeginnMonat" required = "yes"/>
        <sdf:includeVariable variable = "LeistBeginnJahr" required = "yes"/>
        <sdf:includeVariable variable = "GewUrsache1" required = "yes"/>
        <sdf:includeVariable variable = "GewUrsache2" required = "conditional"/>
        <sdf:includeVariable variable = "NettoAnspruch" required = "yes"/>
        <sdf:includeVariable variable = "RegelAnspruch" required = "yes"/>
        <sdf:includeVariable variable = "AnerkAufwend" required = "condi-
tional"/>
        <sdf:includeVariable variable = "KrankBeitrag" required = "condi-
tional"/>
        <sdf:includeVariable variable = "Mehrbedarf" required = "conditional"/>
        <sdf:includeVariable variable = "KeinEink" required = "conditional"/>
        <sdf:includeVariable variable = "ErwerbsEink" required = "conditional"/>
        <sdf:includeVariable variable = "RenteErwerbsMind" required = "condi-
tional"/>
        <sdf:includeVariable variable = "RenteAlter" required = "conditional"/>
        <sdf:includeVariable variable = "RenteHinterblieb" required = "condi-
tional"/>
        <sdf:includeVariable variable = "RentePrivAltVorsorge" required = "con-
ditional"/>
        <sdf:includeVariable variable = "RenteSonstVorsorge" required = "condi-
tional"/>
        <sdf:includeVariable variable = "VersorgBezuege" required = "condi-
tional"/>
        <sdf:includeVariable variable = "RenteBetrAltVorsorge" required = "con-
ditional"/>
        <sdf:includeVariable variable = "LeistGKV" required = "conditional"/>
        <sdf:includeVariable variable = "Kindergeld" required = "conditional"/>
        <sdf:includeVariable variable = "Wohngeld" required = "conditional"/>
        <sdf:includeVariable variable = "PrivUnterhalt" required = "condi-
tional"/>
        <sdf:includeVariable variable = "EinkEhe" required = "conditional"/>
        <sdf:includeVariable variable = "SonstEink" required = "conditional"/>
    </sdf:group>

    <sdf:group attachment-level="message" required="yes">

```

```
        <sdf:includeVariable variable = "ErhStelleGemeinde" required = "yes"/>
        <sdf:includeVariable variable = "KennNrInGemeinde" required = "condi-
tional"/>
    </sdf:group>

</sdf:outputDataModel>

</sdf:surveyDefinition>
```

10. XML-Schema 1.0 *[nicht-normativ]*

Dieser Ausdruck des XML-Schemas kann nicht als Referenz verwendet werden.

```
<?xml version = "1.0" encoding = "ISO-8859-1"?>
<xs:schema xmlns = "http://www.destatis.de/schema/datml-sdf/1.0"
  targetNamespace = "http://www.destatis.de/schema/datml-sdf/1.0"
  xmlns:xs = "http://www.w3.org/2001/XMLSchema"
  xmlns:ssp = "http://www.destatis.de/schema/datml-ssp/1.0"
  version = "1.0"
  elementFormDefault = "qualified">

  <xs:annotation>
    <xs:documentation xml:lang = "de" source = "http://www.statspez.de">
      <![CDATA[
        Schema      : DatML/SDF
        Version     : 1.0
        Variante    : D
        Edition     : 0
        Typ         : XML-Schema
        Namensraum : http://www.destatis.de/schema/datml-sdf/1.0
        Stand      : 20041213
        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/SDF beschreibt Erhebungen und die zu liefernden Merkmale so, dass voll-
        ständige und valide Rohdatenlieferungen einmalig oder über längere Zeiträume
        automatisch erzeugt werden können. Verbraucher von DatML/SDF sind z.B.
        Anwendungen des betrieblichen Rechnungswesens.

        Verwendung

        DatML/SDF (Data Markup Language - Survey Definition Format) unterstützt mit
        seiner umfassenden formalen Beschreibung der Eigenschaften und Merkmale einer
        Erhebung die Automatisierung der Datengewinnungsprozesse in der Unternehmens-DV
        und die Erzeugung der Lieferdokumente. Der Einsatz effizienter generischer
        Methoden erleichtert dabei die Aktualisierung der Verfahren und gewährleistet
        eine hohe Datenqualität.

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

        Updates (U) und Korrekturen (K):

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

  <!-- Inclusion of DatML/SSP -->

  <xs:import namespace = "http://www.destatis.de/schema/datml-ssp/1.0"
    schemaLocation = "datml-ssp-frame-1.xsd"/>

  <!-- Import Definitons from the xml namespace -->

  <xs:import namespace = "http://www.w3.org/XML/1998/namespace"
    schemaLocation = "xml.xsd"/>

  <!-- Beginn DatML/SDF -->

  <!-- Root element -->

  <xs:element name = "surveyDefinition">
    <xs:complexType>
      <xs:sequence>
```

```

        <xs:element ref = "ssp:properties" minOccurs = "0"/>
        <xs:element ref = "ssp:objectContext" minOccurs = "0"/>
        <xs:element ref = "ssp:context"/>
        <xs:element ref = "dataModel"/>
        <xs:element ref = "outputDataModel" minOccurs = "0"/>
        <xs:element ref = "referenceData" minOccurs = "0"/>
        <xs:element ref = "utilities" minOccurs = "0"/>
    </xs:sequence>
    <xs:attribute name = "version" use = "required">
        <xs:simpleType>
            <xs:restriction base = "xs:NMTOKEN">
                <xs:enumeration value = "1.0"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
</xs:complexType>
</xs:element>
<xs:element name = "parameters">
    <xs:complexType/>
</xs:element>
<xs:element name = "dataModel">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref = "variables"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name = "outputDataModel">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref = "group" maxOccurs = "unbounded"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name = "referenceData">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref = "ssp:classification" minOccurs = "0" maxOccurs
= "unbounded"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name = "utilities">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref = "parameters" minOccurs = "0"/>
            <xs:element ref = "adapters" minOccurs = "0"/>
            <xs:element ref = "filters" minOccurs = "0"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name = "variables">
    <xs:complexType>
        <xs:sequence>
            <xs:choice maxOccurs = "unbounded">
                <xs:element ref = "ssp:variable"/>
                <xs:element ref = "ssp:variableGroup"/>
            </xs:choice>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name = "adapters">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref = "adapter" maxOccurs = "unbounded"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name = "adapter">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref = "ssp:simpleType"/>
            <xs:element ref = "ssp:valueSpace"/>
            <xs:element ref = "ssp:description" minOccurs = "0"/>
            <xs:element ref = "ssp:annotation" minOccurs = "0"/>
        </xs:sequence>
        <xs:attribute name = "name" use = "required" type = "xs:NMTOKEN"/>
    </xs:complexType>

```

```

</xs:element>
<xs:element name = "filters">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref = "filter" maxOccurs = "unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name = "filter">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref = "ssp:value"/>
      <xs:element ref = "ssp:description" minOccurs = "0"/>
      <xs:element ref = "ssp:annotation" minOccurs = "0"/>
    </xs:sequence>
    <xs:attribute name = "adapter" use = "required" type = "xs:NMTOKEN"/>
    <xs:attribute name = "name" use = "required" type = "xs:NMTOKEN"/>
  </xs:complexType>
</xs:element>
<xs:element name = "includeVariable">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref = "setValue" minOccurs = "0"/>
    </xs:sequence>
    <xs:attribute name = "variable" use = "required" type = "xs:NMTOKEN"/>
    <xs:attribute name = "required">
      <xs:simpleType>
        <xs:restriction base = "xs:NMTOKEN">
          <xs:enumeration value = "yes"/>
          <xs:enumeration value = "no"/>
          <xs:enumeration value = "conditional"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name = "conditions" type = "xs:NMTOKENS"/>
    <xs:attribute name = "filters" type = "xs:NMTOKENS"/>
  </xs:complexType>
</xs:element>
<xs:element name = "includeVariableGroup">
  <xs:complexType>
    <xs:attribute name = "variable-group" use = "required" type =
"xs:NMTOKEN"/>
    <xs:attribute name = "index" default = "automatic">
      <xs:simpleType>
        <xs:restriction base = "xs:NMTOKEN">
          <xs:enumeration value = "automatic"/>
          <xs:enumeration value = "by-variable"/>
          <xs:enumeration value = "undefined"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name = "index-variable" type = "xs:NMTOKEN"/>
    <xs:attribute name = "required">
      <xs:simpleType>
        <xs:restriction base = "xs:NMTOKEN">
          <xs:enumeration value = "yes"/>
          <xs:enumeration value = "no"/>
          <xs:enumeration value = "conditional"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name = "conditions" type = "xs:NMTOKENS"/>
    <xs:attribute name = "filters" type = "xs:NMTOKENS"/>
  </xs:complexType>
</xs:element>
<xs:element name = "setValue">
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base = "xs:string">
        <xs:attribute name = "mutable" default = "no">
          <xs:simpleType>
            <xs:restriction base = "xs:NMTOKEN">
              <xs:enumeration value = "yes"/>
              <xs:enumeration value = "no"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>

```

```
        </xs:simpleContent>
      </xs:complexType>
    </xs:element>
    <xs:element name = "group">
      <xs:complexType>
        <xs:sequence>
          <xs:choice maxOccurs = "unbounded">
            <xs:element ref = "includeVariable"/>
            <xs:element ref = "includeVariableGroup"/>
          </xs:choice>
        </xs:sequence>
        <xs:attribute name = "attachment-level" use = "required">
          <xs:simpleType>
            <xs:restriction base = "xs:NMTOKEN">
              <xs:enumeration value = "data"/>
              <xs:enumeration value = "message"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
        <xs:attribute name = "required">
          <xs:simpleType>
            <xs:restriction base = "xs:NMTOKEN">
              <xs:enumeration value = "yes"/>
              <xs:enumeration value = "no"/>
              <xs:enumeration value = "conditional"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
        <xs:attribute name = "conditions" type = "xs:NMTOKENS"/>
        <xs:attribute name = "filters" type = "xs:NMTOKENS"/>
        <xs:attribute name = "name" type = "xs:string"/>
      </xs:complexType>
    </xs:element>

    <!-- Ende DatML/SDF -->

  </xs:schema>
```