
Spezifikation XDatenfelder

Version 3.0.0

Fassung: 25.06.2021

Herausgeber: FIM Baustein Datenfelder

Namensraum: urn:xoev-de:fim:standard:xdatenfelder_3.0.0

Bezugsort: www.xrepository.de

Inhaltsverzeichnis

I Überblick	1
Vorwort	3
Einleitung	5
Kommunikationsszenarien	5
Aufbau der Spezifikation	7
Bestandteile des Standards	7
II Allgemeines	9
II.1 Grundlegende Begriffe	11
II.1.1 Glossar	11
II.1.2 Abkürzungsverzeichnis	13
II.2 Basisdatentypen	15
II.2.1 AnzahlString	15
II.2.2 CountryCode	15
II.2.3 Language	15
II.2.4 LanguageVariant	15
II.2.5 PatternString	15
II.2.6 VersionString	16
II.3 Das Informationsmodell	17
II.3.1 Baukasten	17
II.3.2 Codes und Codelisten	41
II.A Eingebundene externe Modelle	45
II.A.1 XOEV-Bibliothek	45
III Datenübermittlung	47
III.1 Übertragung von Inhalten	49
III.1.1 Die Nachrichten	49
IV Anhänge	55
IV.A Übersicht über alle Nachrichten	57
IV.B Die Codelisten des Standards XDatenfelder	59
IV.B.1 Details	59
IV.C Mitwirkende	69
IV.D Versionshistorie	71
Stichwortverzeichnis	73



I Überblick

Vorwort



Das Projekt "Föderales Informationsmanagement" (FIM) verfolgt das Ziel, Informationen zu Verwaltungsvorgängen zwischen Bund, Ländern und Kommunen auf freiwilliger Basis zu harmonisieren. Ein wesentlicher Bestandteil des Projekts ist die Harmonisierung von Datenstrukturen und Logik für Formulare. Formulare sind strukturierte Sammlungen von Informationsobjekten, die der Verwaltung die standardisierte und direkte Erfassung/Bereitstellung von Daten ermöglichen. Mögliche Ausprägungen solcher Informationsobjekte sind: Antragsformulare, Bescheide oder andere externe Dokumente.

Diese Aufgabe übernimmt der FIM Baustein Datenfelder. Er enthält unter anderem die folgenden Elemente Datenfeld, Datenfeldgruppe, Regel und Codeliste. Das Datenfeld bildet die kleinste Einheit des FIM-Baukastens für den Baustein Datenfelder. Eine Datenfeldgruppe besteht aus einer Sammlung von Datenfeldern und/oder Datenfeldgruppen sowie weiteren datenfeldgruppenspezifischen Informationen. Logische Abhängigkeiten zwischen Datenfeldern sowie zwischen Datenfeld und Datenfeldgruppe können durch Regeln ausgestaltet werden. Eine Codeliste ist eine Liste von vordefinierten Werten, die Felder annehmen können. Alle Baukastenelemente werden definiert und ein dazugehörigen Satz an Metadaten vorgeschlagen, welcher notwendig ist, um die Elemente zielführend zu nutzen.

Die FIM-Elemente Dokumentsteckbrief sowie Stammdatenschema konzipiert. Dokumentsteckbriefe dienen einer allgemeinen Beschreibung von Formularen, indem Metadaten verwendet werden, um möglichst stabile und über die Zeit wenig veränderliche Angaben zu einem Formular festzuhalten. Solche Metadaten wären zum Beispiel die Beschreibung und die Rechtsgrundlagen des Formulars. Somit dienen Dokumentsteckbriefe als Grundlage für die Erstellung von Stammdatenschemata. Stammdatenschemata sind die höchste Aggregationsstufe der Informationen im Baustein Datenfelder. Zu jedem Stammdatenschema, welches in einer FIM-Bibliothek abgelegt wird, werden Metadaten erfasst bzw. gepflegt. Über die Metadaten werden Informationen zu dessen Inhalt sowie auch strukturellen, redaktionellen und technischen Eigenschaften bereitgestellt. Die Metadaten dienen im Rahmen der Befüllung und Nutzung der FIM-Bibliothek der besseren Strukturierung und Klassifizierung von Stammdatenschemata, zur strukturierten Ablage und besseren Auffindbarkeit sowie zur Einbindung der Stammdatenschemata in den FIM-Gesamtkontext.

Die vom Baustein Datenfelder erarbeiteten Ergebnisse werden im Standard XDatenfelder zur Nachnutzung im FIM-Portal, zur Synchronisation zwischen Repositorys und Nutzung durch nachgelagerte Systeme bereitgestellt. Das vorliegende Dokument beschreibt den Standard vollständig und XÖV-konform.

Einleitung



Der Standard XDatenfelder definiert ein Austauschformat für die vom FIM-Baustein Datenfelder bereitgestellten Inhalte.

Kommunikationsszenarien

Das XÖV-Regelwerk sieht vor, dass alle Dokumente als Nachrichten spezifiziert werden. Um die Liste der benötigten Nachrichten aufzustellen, werden in diesem Kapitel die Einsatzszenarien für den Standard XDatenfelder betrachtet und daraus die benötigten Nachrichten abgeleitet.

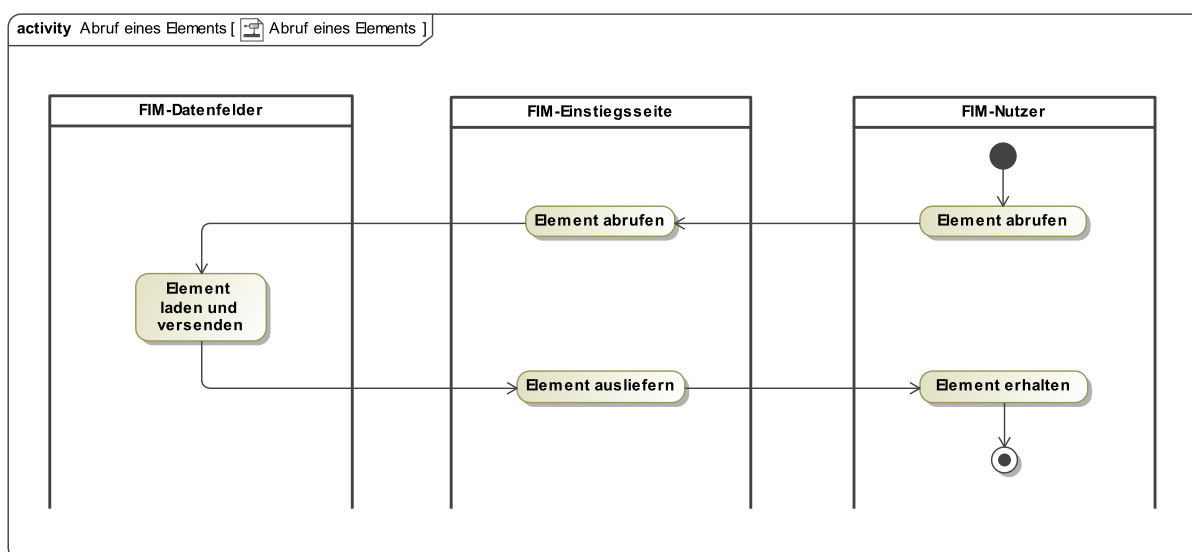
1.1 FIM-Einstiegsseite

Es werden Dokumentsteckbriefe und Stammdatenschemata über die FIM-Einstiegsseite zur Verfügung gestellt.

Benötigte Nachrichten:

- Dokumentsteckbrief übertragen
- Stammdatenschema übertragen
- Datenfeldgruppe übertragen
- Datenfeld übertragen
- Codeliste übertragen

Abbildung 1. Abruf eines Elements



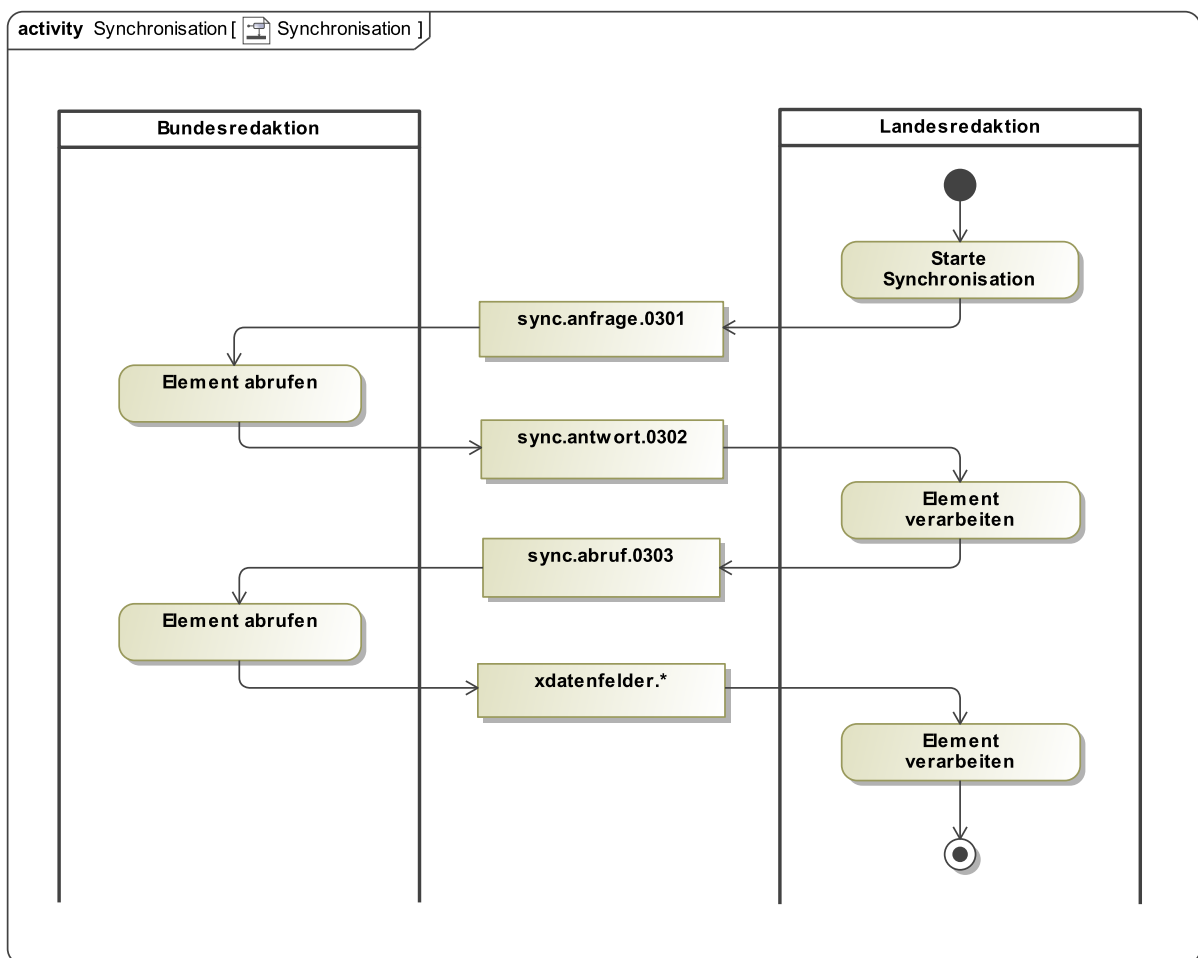
1.2 Landesredaktionen

Zur Anbindung der Landesredaktionen müssen Teile des Datenfeldbaukastens der Bundesredaktion an die Landesredaktionen übertragen werden.

Benötigte Nachrichten:

- Abruf aller neuen Elemente - Anfrage
- Abruf aller neuen Elemente - Antwort
- Element abrufen

Abbildung 2. Synchronisation der Landesredaktionen



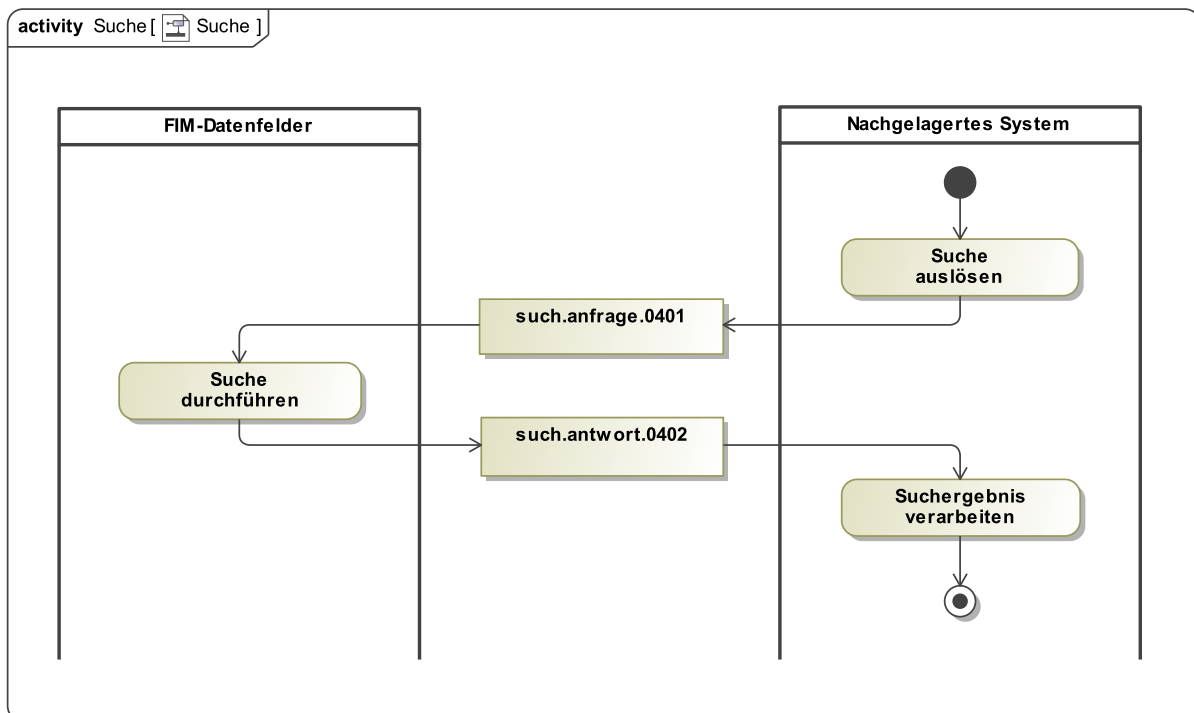
1.3 Nachgelagerte Systeme

Nachgelagerte Systeme, wie z.B. an Landesredaktionssysteme angebundene Antragsmanagementsysteme sollen Suchanfragen an die Redaktionssysteme stellen können.

Benötigte Nachrichten:

- Suchanfrage
- Suchantwort

Abbildung 3. Suche im Redaktionssystem



Aufbau der Spezifikation

Im Kapitel "Allgemeines" werden zuerst die Bausteine beschrieben, aus denen sich die Nachrichten zusammensetzen, welche im folgenden Kapitel "Datenübermittlung" dokumentiert sind.

Bestandteile des Standards

Bestandteile des Standards sind:

- **Spezifikationsdokument:** Dieses Spezifikationsdokument
- **XML Schema-Definitionen:** Die technische Spezifikation als XML Schema (XSD)



II Allgemeines

II.1 Grundlegende Begriffe



II.1.1 Glossar

Tabelle II.1.1. Glossar

Begriff	Erläuterung
FIM-Baukasten (Leistungsbaukasten, Datenfelderbaukasten, Prozessbaukasten)	Strukturierte Sammlung von standardisierten FIM-Elementen, wie z. B. Datenfeldgruppen, Datenfeldern oder LeiKa-Modulen, aus denen Stamminformationen wie Stammdatenschemata, Stammtexte oder Stammprozesse zusammengestellt werden können. Er dient der redaktionellen Standardisierung von Leistungsberichten, Formularen und Prozessen.
FIM-Bibliothek (Stamtextbibliothek, Stammdatenfelderbibliothek, Stammprozessbibliothek)	Strukturierte Sammlung von Stamminformationen, d.h. Stammtexten, Stammdatenschemata und Stammprozessen. Die Stamminformationen werden eindeutig durch Schlüsselnummern identifiziert. Bezüge zwischen Stamminformationen, z.B. zwischen Stammtext, Stammdatenfeldern und dem entsprechenden Stammprozess werden durch den FIM-Ordnungsrahmen abgebildet und durch die Systematik der Schlüsselnummern nachvollziehbar.
FIM-Element	Standardisierte Teile von Leistungsberichten, Formularen oder Prozessen, wie z. B. Datenfeldgruppen oder Prozessbausteine, die zur Erstellung von Stamminformationen genutzt werden. FIM-Elemente können auch direkt für die Erstellung von Formularen, Leistungsbeschreibungen oder Prozessen genutzt werden, auch wenn nicht die Bereitstellung einer Stamminformation als Ziel angestrebt wird. Ein Beispielfall dafür ist die Erstellung von Formularen für individuelle kommunale Leistungen, wie z. B. Familienrabattkarten für reduzierte Eintrittspreise in öffentlichen Einrichtungen.
FIM-Ergebnis	Alle Ergebnisse des Projekts FIM, die veröffentlicht und den Nutzern zur Verfügung gestellt werden. Die FIM-Ergebnisse umfassen insbesondere alle Bestandteile des FIM-Hauses, also die FIM-Logik, die FIM-Werkzeuge, das FIM-Redaktionsmodell, die FIM-XÖV-Standards, die FIM-Software-Systeme sowie die FIM-Elemente (FIM-Baukastenelemente und FIM-Stamminformationen).
FIM-Kataloge (Leistungskatalog, Datenfelder katalog, Prozesskatalog)	Die drei FIM-Kataloge dienen dazu, identifizierende Beschreibungen zu den Stamminformationen aus den drei FIM-Bausteinen zu erfassen und ihnen eine bundesweit eindeutige Identifikationsnummer zuzuordnen. FIM-Kataloginformationen sind Leistungssteckbriefe, Dokumentsteckbriefe und Prozessklassen.
FIM-Nutzer	Nutzt FIM-Inhalte über FIM-Systeme bzw. nachgelagerte Systeme und formuliert ggf. Änderungs- und Erweiterungsbedarfe.
Formular	Das Formular ist ein Instrumentarium der öffentlichen Verwaltung, mit dem einerseits Daten für verschiedene Zwecke erhoben und Informationen bspw. zur Bearbeitung von Anträgen auf Leistungen unterschiedlicher Art nachgefragt werden können (Becker 2009). Beispiele hierfür sind u.a. Formulare für die polizeiliche Anmeldung, Beantragungen von Baugenehmigungen sowie Wohngeldzuschuss. Neben diesen reinen Input-Dokumenten werden ebenfalls Output-Dokumente, wie Bescheide oder invariante Dokumente, wie Ausweise unter dem Formularbegriff zusammengefasst. Auf der anderen Seite bezeichnen Formulare im FIM-Kontext nicht nur Informationen an den Schnittstellen der Verwaltungsprozesse, d.h. Auslöser und Ergebnisse, sondern umfassen auch die Informationen, die 'indirekt' für einen Prozess relevant

Begriff	Erläuterung
	sind und im Rahmen einer Leistungserstellung verarbeitet werden. Eine Ausprägung dieses Formularverständnisses stellt der LeiKa dar, der mithilfe seiner Modulinformationen die Eingangsinformationen für verschiedene nachgelagerte Prozesse liefert. Formulare sind strukturierte Sammlungen von Informationsobjekten, die der Verwaltung die standardisierte und direkte Erfassung/Bereitstellung von Daten ermöglichen. In der Regel, geben Formulare spezifische Formularfelder vor, in die Informationen unterschiedlicher Art eingefügt werden können. Typische Feldvarianten sind Textfelder (zum Beispiel "Nachname", "Vorname", "Adresse") und Einfach- (zum Beispiel "verheiratet: ja/nein") oder Mehrfachauswahlfelder. Formulare können sowohl papierbasierte Dokumente als auch Schemavorgaben für die Nutzung elektronischer Schnittstellen sein. Formulare werden im FIM-Baustein Datenfelder abgebildet. Er bildet zusammen mit Leistungen und Prozessen die drei Bausteine von FIM.
Datenfeld	Unter einem (Daten-)Feld wird grundsätzlich die kleinste logische Einheit eines Datensatzes oder eines Formulars zur Datenerfassung verstanden (Stahlknecht, Hasenkamp, 2002). Im Kontext des FIM-Bausteins Datenfelder bezeichnen Datenfelder die elementaren Bestandteile zur Beschreibung von Datenstrukturen und stellen einen Typ von Stamminformationen dar. Bei der Definition von Datenfeldern werden durch die regulatorisch zuständigen Stellen Name, Definition und ein Identifikationsschlüssel einheitlich vorgegeben. Wertebereiche, Hilfetexte und sonstige Merkmale können auf Vollzugsebene angepasst werden. Typische Kombinationen von Feldern werden zu Feldgruppen zusammengefasst. Kombinationsmöglichkeiten von Feldern und Feldgruppen werden durch Feldregeln abgebildet. Dabei kann ein Feld bzw. eine Feldgruppe in mehreren Stammdatenschemata verwendet werden. Somit kann ein Feld bzw. die damit erhobenen Daten auch in unterschiedlichen Prozessen bzw. Leistungen verwendet werden.
Datenfeldgruppe	Datenfeldgruppen dienen dazu, Datenfelder zu logisch in Beziehung stehenden Gruppen zusammen zu fassen. Datenfeldgruppen beinhalten folglich Datenfelder und ermöglichen gleichzeitig die strukturierte Darstellung von Stammdatenschemata.
Identifizierungsschlüssel	Identifizierungsschlüssel dienen zur Identifizierung eines bestimmten Elements eines Kataloges oder einer Bibliothek. Teile eines Schlüssels repräsentieren ggf. verschiedene Klassifikationsmerkmale. Beziehungen zwischen FIM-Stamminformationen können mit Hilfe der Identifizierungsschlüssel über Bausteine hinweg hergestellt werden.
LeiKa	Mit dem Leistungskatalog (LeiKa) wird in Deutschland ein einheitliches und umfassendes Verzeichnis der Verwaltungsleistungen des Bundes, der Länder und Kommunen bereitgestellt. Ziel ist es, eine zentrale Informationsbasis in Form eines Stammtext-Managements zur Verfügung zu stellen, die von allen Verwaltungsbereichen anwendungs- und vorhabenübergreifend für alle Informations- und Kommunikationskanäle genutzt wird. Weitere Informationen unter: www.fimportal.de .
Leistung	Als Leistung einer öffentlichen Verwaltung (auch Verwaltungsleistung) wird im engeren Sinne ein nach außen für Dritte (z. B. Bürger, Unternehmen, freie oder gemeinnützige Träger) sichtbares Ergebnis der Verwaltungstätigkeit bezeichnet (Schedler, Proeller, 2006; GK LeiKa, 2012). Kennzeichnend für eine Leistung ist, dass es sich um das kleinteiligste Arbeitsergebnis handelt, das in Anspruch genommen werden kann (Spitzer, 1998). Es können verschiedene Arten von Verwaltungsleistungen unterschieden werden. Zum Ersten, können Verwaltungsleistungen auf Antrag erbracht werden, d.h. die Leistung wird nur erbracht, wenn ein Antragssteller einen entsprechenden Antrag stellt. Zum Zweiten, können Verwaltungsleistungen auf Verlangen ohne die Notwendigkeit eines Antrages in Anspruch genommen werden, beispielsweise in Form von Informationsbereitstellungen. Schließlich, gibt es proaktive Verwaltungsleistungen, bei denen die Verwaltung von sich aus tätig wird (von Lucke, 2008). Interne (Teil-)Arbeitsergebnisse stellen nach diesem, engeren Verständnis zunächst keine Leistungen dar. In einem weiteren Verständnis lassen sie sich aber als interne Leistungen charakterisieren, da sie unmittelbar oder mittelbar zur eigentli-

Begriff	Erläuterung
	chen Leistungserstellung beitragen. Leistungen bilden zusammen mit Prozessen und Datenfelder die drei Bausteine von FIM.
Prozess	"Ein Prozess ist die inhaltlich abgeschlossene, zeitliche und sachlogische Folge von Aktivitäten, die zur Bearbeitung eines betriebswirtschaftlich relevanten Objektes notwendig sind" (Becker, Kahn, 2008, S. 6). Unter einem betriebswirtschaftlich relevanten Objekt kann im Kontext von FIM auch eine Verwaltungsleistung (nach Eilwein) verstanden werden. "Ein solches Objekt wird aufgrund seiner zentralen Bedeutung für den Prozess als prozessprägendes Objekt bezeichnet. So wird bspw. der Prozess der Rechnungsprüfung durch das Objekt der Rechnung geprägt" (Becker, 2012). Ein Prozess beschreibt die Art, wie Organisationen arbeiten und ist dabei eine Menge an Aktivitäten, die von diesen Organisationen durchgeführt werden, um ein bestimmtes Ziel zu erreichen (Davenport, 2008). Prozesse transformieren dabei Inputfaktoren zu einem Output Faktor (Krcmar, 2009). "In Bezug auf die öffentliche Verwaltung ist Prozess die kleinste operationalisierte Einheit einer Verwaltungsmaßnahme mit einem abgeschlossenen, aufgabenbezogenen Arbeitsergebnis" (Becker et al., 2007, S.30). Prozesse bilden zusammen mit Leistungen und Datenfelder die drei Bausteine von FIM.
Referenzaktivitätengruppe	In einem Stammprozess fasst eine Aktivitätengruppe gemäß dem geforderten Abstraktionsniveau verschiedene Aktivitäten zusammen. Referenzaktivitätengruppen sind die standardisierten Bauelemente des Prozessbaukastens, aus denen Aktivitätengruppen abgeleitet werden.
Stamminformation (Stammdatenschema, Stammtext, Stammprozess)	Stamminformationen (hier enthalten sind: Stammtexte, Stammdatenschemata und Stammprozesse) werden im Kontext von FIM als standardisierte Beschreibungen der drei Aspekte (Datenstrukturen, Leistungs- und Ablaufinformationen) von Verwaltungsleistungen betrachtet, die von der jeweils regulatorisch zuständigen Verwaltungsstelle erstellt und aktualisiert werden. Diese werden auf die Einhaltung gemeinsamer Qualitätskriterien hin geprüft und im weiteren Verlauf auf Landes- und Kommunalebene bis zu ihrer endgültigen Fertigstellung weiter konkretisiert. Auf der Vollzugsebene werden Stamminformationen verwendet, kombiniert und ggf. um individuelle Bestandteile ergänzt, um konkrete Instanzen von der Beschreibungen von Leistungen, Formularen und Prozessen zu erstellen. Stamminformationen dienen so als standardisierte Referenz für alle konkretisierten Instanzen auf Vollzugsebene.

II.1.2 Abkürzungsverzeichnis

Tabelle II.1.2. Abkürzungsverzeichnis

Abkürzung	Langbezeichnung
BPMN	Business Process Model and Notation
EA 2.0	Projekt Einheitlicher Ansprechpartner 2.0
FIM	Föderales Informationsmanagement
GK	Geschäfts- und Koordinierungsstelle
LeiKa	Leistungskatalog der öffentlichen Verwaltung
XÖV	XML in der öffentlichen Verwaltung

II.2 Basisdatentypen



II.2.1 AnzahlString

Typ: **AnzahlString**

Beschreibt einen String, der die Multiplizität (minimale und maximale Anzahl) der Elemente beschreibt. Beispiele: "0:1", "0:7" oder "1:*".

Dieser Typ ist eine Einschränkung des Basistyps **xs:string**.

Die Werte müssen dem Muster `'d+:(^|\d+)'` entsprechen.

II.2.2 CountryCode

Typ: **CountryCode**

Ländercode, bestehend aus zwei Großbuchstaben gemäß ISO 3166-1 alpha-2. Beispiel: "DE" für Deutschland.

Dieser Typ ist eine Einschränkung des Basistyps **xs:token**.

Die Werte müssen dem Muster `'[A-Z]{2}'` entsprechen.

Beschränkungen (XML-Schema-Facetten): `length:2`

II.2.3 Language

Typ: **Language**

Die Primary Language darf nur aus zwei oder drei Kleinbuchstaben bestehen (gemäß ISO 639-1 (2002), ISO 639-2 (1998), ISO 639-3 (2007) oder ISO 639-5 (2008)).

Dieser Typ ist eine Einschränkung des Basistyps **xs:token**.

Die Werte müssen dem Muster `'[a-z]{2,3}'` entsprechen.

Beschränkungen (XML-Schema-Facetten): `maxLength:3; minLength:2`

II.2.4 LanguageVariant

Typ: **LanguageVariant**

Sprachvariante (gemäß IANA Registry), derzeit wird nur "simple" für einfache Sprache unterstützt.

Dieser Typ ist eine Einschränkung des Basistyps **xs:token**.

Die Werte müssen dem Muster `'simple'` entsprechen.

II.2.5 PatternString

Typ: **PatternString**

Ein Media Type gemäß der IANA Registry (<https://www.iana.org/assignments/media-types/media-types.xhtml>).

Dieser Typ ist eine Einschränkung des Basistyps `xs:token`.

Die Werte müssen dem Muster `'w+/-+.\w+'` entsprechen.

II.2.6 VersionString

Typ: `VersionString`

Ein String, der eine Versionsnummer darstellt. Seit Version 3.0 ist dieser dreiteilig. Beispiel: "1.0.0" oder "2.7.3".

Dieser Typ ist eine Einschränkung des Basistyps `xs:string`.

Die Werte müssen dem Muster `'d+.\d+.\d+'` entsprechen.

II.3 Das Informationsmodell



II.3.1 Baukasten

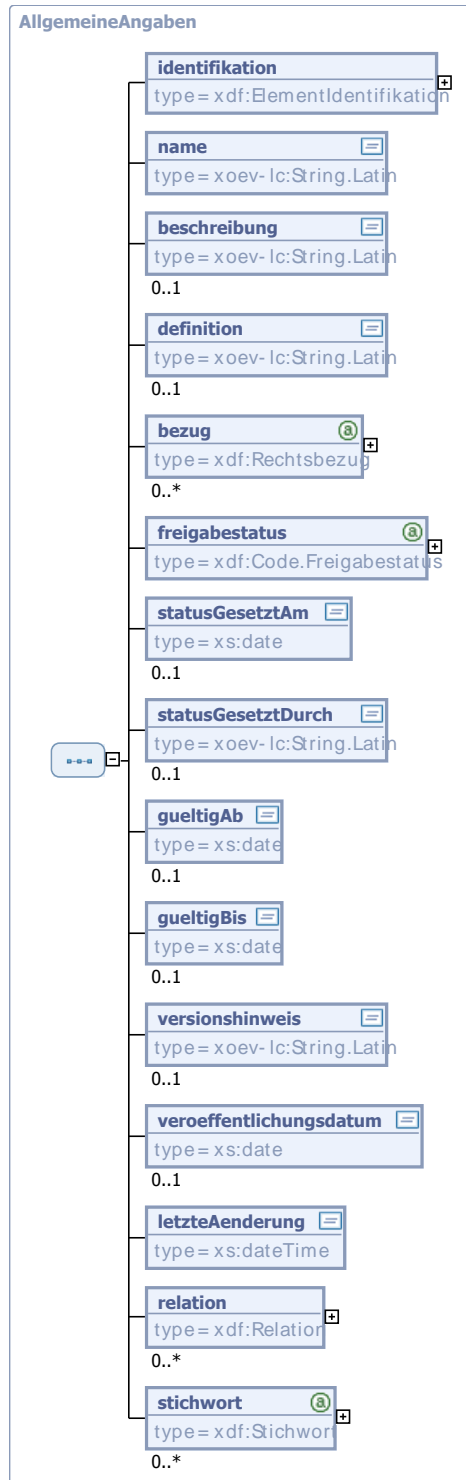
In diesem Kapitel werden die Bausteine beschrieben, aus denen sich im Folgenden die Nachrichten aufbauen.

II.3.1.1 AllgemeineAngaben

Typ: **AllgemeineAngaben** (abstrakt)

Allgemeine Angaben, die in allen Bausteinen enthalten sind.

Abbildung II.3.1. AllgemeineAngaben



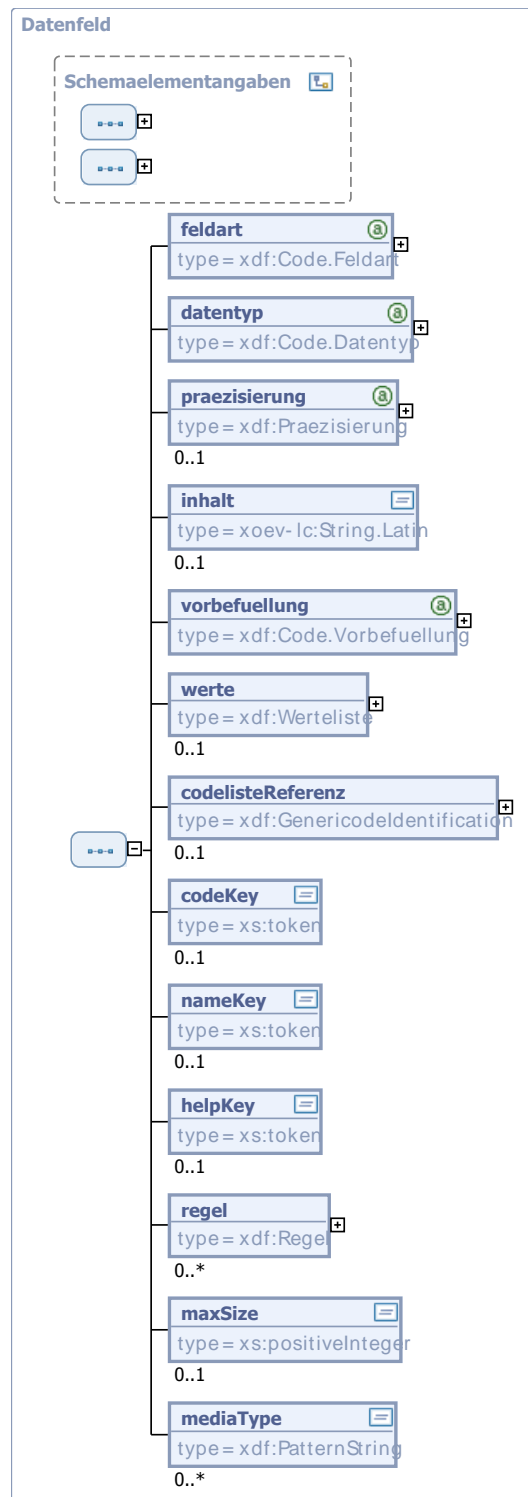
Kindelemente von AllgemeineAngaben				
Kindelement	Typ	Anz.	Ref.	Seite
identifikation	ElementIdentifikation	1	II.3.1.5	24
Eine eindeutige Identifikation, die aus einer alphanumerischen und fortlaufenden Nummerierung bestehenden ID (Beispiel: "F0000000001") und der Versionsnummer zusammengesetzt ist.				
name	String.Latin	1	II.A.1	45
FIM-interner Name des Elements.				
beschreibung	String.Latin	0..1	II.A.1	45
Zusätzliche Beschreibung und Erläuterung des Elements für FIM- Nutzer.				
definition	String.Latin	0..1	II.A.1	45
Fasst den Inhalt des Elements aus redaktioneller Sicht zusammen, d.h. aus Sicht der Rollen fachlicher Ersteller, Methodenexperte, Informationsmanager, FIM-Koordinierungsstelle, FIM-Nutzer.				
bezug	Rechtsbezug	0..n	II.3.1.14	31
Bezug zu Rechtsnorm oder Standardisierungsvorhaben: Dieses Attribut ermöglicht es, Bezüge zu einschlägigen Rechtsnormen sowie zu in anderen Vorhaben wie den XÖV-Vorhaben (z.B. XWaffe) oder Vorhaben wie P23R gleich oder ähnlich definierten Datencontainern zu erstellen und diese Verbindung zu dokumentieren. Hierdurch lässt sich die Ausgestaltung eines Stammdatenschemas gemäß einschlägigen Rechtsnormen oder Standardisierungsverfahren belegen.				
freigabestatus	Code.Freigabestatus	1	II.3.2.7	42
Der Status gibt Aufschluss darüber, ob und wie das Element im Rahmen des Editors zu verwenden ist. Mögliche Ausprägungen dieses Metadatums sind: "aktiv", "in Vorbereitung", "inaktiv".				
statusGesetztAm	xs:date	0..1		
Datum der Freigabe durch die Freigabeinstanz.				
statusGesetztDurch	String.Latin	0..1	II.A.1	45
Ersteller oder erstellende Institution.				
gueltigAb	xs:date	0..1		
Sofern es eine Beschränkung der Gültigkeit gibt, ist der Gültigkeitszeitraum zu erfassen. Mit diesem Metadatum wird der Beginn der Gültigkeit erfasst.				
gueltigBis	xs:date	0..1		
Sofern es eine Beschränkung der Gültigkeit gibt, ist der Gültigkeitszeitraum zu erfassen. Mit diesem Metadatum wird das Ende der Gültigkeit erfasst.				
versionshinweis	String.Latin	0..1	II.A.1	45
Versionsspezifischer Hinweis.				
veroeffentlichungsdatum	xs:date	0..1		
Datum, an dem das Element veröffentlicht wurde.				
letzteAenderung	xs:dateTime	1		
Datum der letzten Änderung. Wird vom FIM-System automatisch gepflegt.				
relation	Relation	0..n	II.3.1.19	35
Relationen dokumentieren Beziehungen der Elemente untereinander, wie z.B. "ist abgeleitet von", "ersetzt" oder "ist äquivalent zu".				
stichwort	Stichwort	0..n	II.3.1.23	39
Stichwörter sind einerseits frei wählbare Begriffe, die den Elementen zugeordnet werden. Sie können jedoch auch eine Kategorie referenzieren und Werte aus dieser zuordnen.				

II.3.1.2 Datenfeld

Typ: **Datenfeld**

Die Feldart gibt die grundsätzliche Art des Feldes an.

Abbildung II.3.2. Datenfeld



Dieser Typ ist eine Erweiterung des Basistyps `schemaelementangaben` (siehe [Abschnitt II.3.1.20 auf Seite 36](#)).

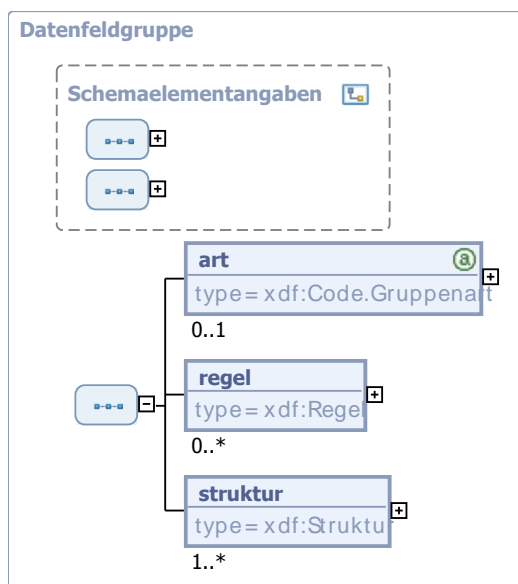
Kindelemente von Datenfeld				
Kindelement	Typ	Anz.	Ref.	Seite
feldart	<code>Code.Feldart</code>	1	II.3.2.6	42
Art des Datenfelds.				
datentyp	<code>Code.Datentyp</code>	1	II.3.2.4	42
Der Datentyp bezeichnet die Zusammenfassung konkreter Wertebereiche und ggf. darauf definierten Operationen zu einer Einheit, die mithilfe von Regeln abbildbar sind. Zudem spezifiziert der Datentyp die Struktur des Inhalts oder gibt Aufschlüsse über die Feldinhaltsänderbarkeit.				
praezisierung	<code>Praezisierung</code>	0..1	II.3.1.13	30
Mit der Präzisierung kann der Datentyp weiter eingeschränkt werden. Es sind Beschränkungen der Länge und des Wertebereichs möglich.				
inhalt	<code>String.Latin</code>	0..1	II.A.1	45
Dieses Metadatum dient dazu, den Inhalt eines Formularfeldes zu spezifizieren, d.h. ein vordefinierter (Default-)Wert für ein Textfeld.				
vorbefuellung	<code>Code.Vorbefuellung</code>	1	II.3.2.13	43
Gibt an, ob die Behörde das Feld vorbefüllen kann oder muss.				
werte	<code>Werteliste</code>	0..1	II.3.1.26	40
Eine Liste von Werten, die das Feld annehmen kann.				
codelisteReferenz	<code>GenericcodeIdentification</code>	0..1	II.3.1.10	28
Referenz auf eine Codeliste, deren Werte das Feld annehmen kann.				
codeKey	<code>xs:token</code>	0..1		
Bei Verwendung einer Codeliste kann mit diesem Metadatum die zu verwendene Code-Spalte vorgegeben werden.				
nameKey	<code>xs:token</code>	0..1		
Bei Verwendung einer Codeliste kann mit diesem Metadatum die zu verwendene Namens-Spalte (anzuweisender Text) vorgegeben werden.				
helpKey	<code>xs:token</code>	0..1		
Bei Verwendung einer Codeliste kann mit diesem Metadatum die zu verwendene Hilfe-Spalte vorgegeben werden.				
regel	<code>Regel</code>	0..n	II.3.1.15	31
Die Regeln schränken die möglichen Verwendungen des Feldes mittels Skripten ein.				
maxSize	<code>xs:positiveInteger</code>	0..1		
Diese Eigenschaft darf nur bei Datenfeldern des Typs "file" verwendet werden. Sie legt die Maximalgröße in Byte fest.				
mediaType	<code>PatternString</code>	0..n	II.2.5	15
Diese Eigenschaft darf nur bei Datenfeldern des Typs "file" verwendet werden. Sie listet die erlaubten Dateierarten gemäß der Media Type Registry der IANA auf. Beispiel: "application/pdf" für PDF				

II.3.1.3 Datenfeldgruppe

Typ: **Datenfeldgruppe**

Eine Datenfeldgruppe besteht aus einer Sammlung von Datenfeldern und/oder Datenfeldgruppen.

Abbildung II.3.3. Datenfeldgruppe



Dieser Typ ist eine Erweiterung des Basistyps `Schemaelementangaben` (siehe [Abschnitt II.3.1.20](#) auf [Seite 36](#)).

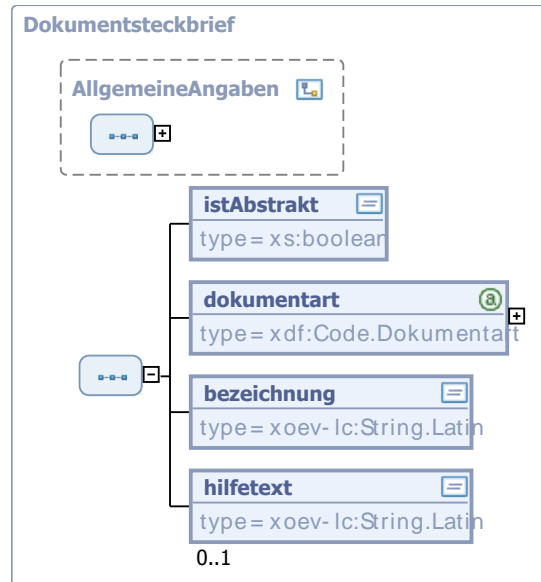
Kindelemente von Datenfeldgruppe				
Kindelement	Typ	Anz.	Ref.	Seite
art	<code>Code.Gruppenart</code>	0..1	II.3.2.8	43
Mit diesem Element werden Datenfeldgruppen mit spezieller Semantik gekennzeichnet.				
regel	<code>Regel</code>	0..n	II.3.1.15	31
Die Regeln schränken die möglichen Verwendungen der Gruppe mittels Skripten ein.				
struktur	<code>Struktur</code>	1..n	II.3.1.24	39
Die Struktur listet alle Unterelemente auf.				

II.3.1.4 Dokumentsteckbrief

Typ: **Dokumentsteckbrief**

Mit einem Dokumentsteckbrief kann im ersten Schritt ein Dokument (Formular) (mit seinen Metadaten) in der Datenfelderbibliothek erfasst werden, ohne gleichzeitig oder direkt anschließend die Struktur des Dokuments (d.h. das zugehörige Stammdatenschema) erstellen zu müssen.

Abbildung II.3.4. Dokumentsteckbrief



Dieser Typ ist eine Erweiterung des Basistyps **AllgemeineAngaben** (siehe [Abschnitt II.3.1.1 auf Seite 17](#)).

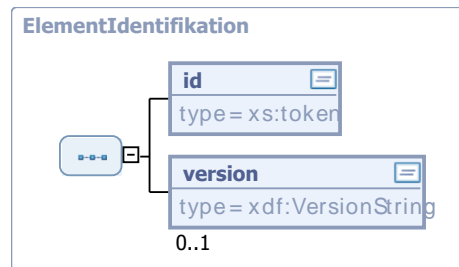
Kindelemente von Dokumentsteckbrief				
Kindelement	Typ	Anz.	Ref.	Seite
istAbstrakt	<code>xs:boolean</code>	1		
Gibt an, ob es sich um ein abstrakten Dokumentsteckbrief handelt.				
dokumentart	<code>Code.Dokumentart</code>	1	II.3.2.5	42
Gibt die Art des Dokuments an.				
bezeichnung	<code>String.Latin</code>	1	II.A.1	45
Bezeichnung des Dokumentsteckbriefs, welches für den Bürger/das Unternehmen auf Input-Formularen sichtbar ist.				
hilfetext	<code>String.Latin</code>	0..1	II.A.1	45
Gibt den Hilfetext für dieses Element.				

II.3.1.5 ElementIdentifikation

Typ: **ElementIdentifikation**

Eindeutige Identifikation eines FIM-Bausteins, bestehend aus der statischen ID und der Versionsnummer.

Abbildung II.3.5. ElementIdentifikation



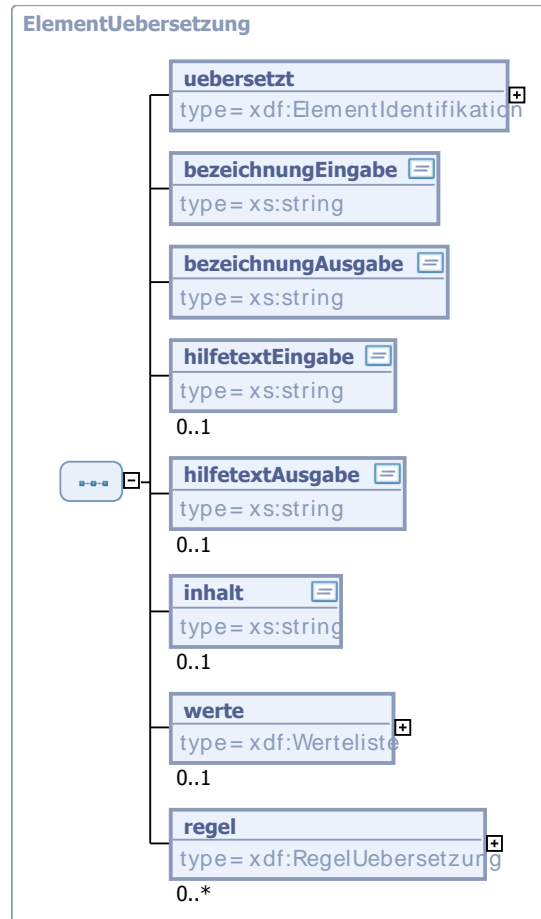
Kindelemente von ElementIdentifikation				
Kindelement	Typ	Anz.	Ref.	Seite
id	<code>xs:token</code>	1		
Statischer ID-Bestandteil der eindeutigen Identifikation eines Elements.				
version	<code>VersionString</code>	0..1	II.2.6	16
Versions-Bestandteil der eindeutigen Identifikation eines Elements.				

II.3.1.6 ElementUebersetzung

Typ: `ElementUebersetzung`

Dieses Element enthält die Übersetzung eines Elements, z.B. eines Datenfelds.

Abbildung II.3.6. ElementUebersetzung



Kindelemente von ElementUebersetzung				
Kindelement	Typ	Anz.	Ref.	Seite
uebersetzt	ElementIdentifikation	1	II.3.1.5	24
Referenz auf das übersetzte Element.				
bezeichnungEingabe	xs:string	1		
Übersetzung der Eigenschaft "bezeichnungEingabe".				
bezeichnungAusgabe	xs:string	1		
Übersetzung der Eigenschaft "bezeichnungAusgabe".				
hilfertextEingabe	xs:string	0..1		
Übersetzung der Eigenschaft "hilfertextEingabe".				
hilfertextAusgabe	xs:string	0..1		
Übersetzung der Eigenschaft "hilfertextAusgabe".				
inhalt	xs:string	0..1		
Übersetzung des vorgegeben Inhalts (fester oder vorgegebener Text).				
werte	Werteliste	0..1	II.3.1.26	40

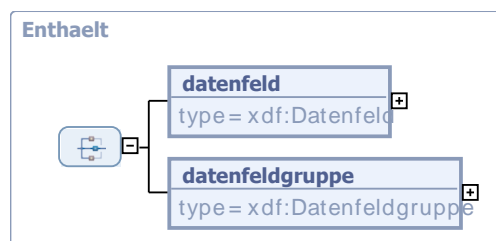
Kindelemente von ElementÜbersetzung				
Kindelement	Typ	Anz.	Ref.	Seite
Übersetzung eine Werteliste. Die Werte der "code" Spalte müssen exakt mit der deutschen Werteliste übereinstimmen. Die "name" und "hilfe" Spalten enthalten die Übersetzung.				
regel	RegelÜbersetzung	0..n	II.3.1.17	34
Übersetzung aller Fehlermeldungen der im Element enthaltenen Regeln.				

II.3.1.7 Enthaeilt

Typ: **Enthaeilt**

Abstrakte Basisklasse, um in die Struktur verschiedene Unterelemente einsetzen zu können (entspricht dem XSD Choice).

Abbildung II.3.7. Enthaeilt



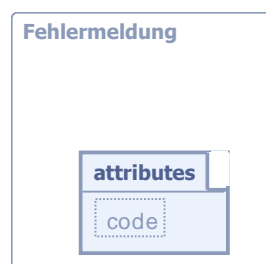
Kindelemente von Enthaeilt				
Kindelement	Typ	Anz.	Ref.	Seite
datenfeld	Datenfeld	1	II.3.1.2	20
Ein Datenfeld als konkrete Instanz.				
datenfeldgruppe	Datenfeldgruppe	1	II.3.1.3	22
Eine Datenfeldgruppe als konkrete Instanz.				

II.3.1.8 Fehlermeldung

Typ: **Fehlermeldung**

Enthält die Fehlermeldung einer Regel. Das Element enthält entweder die ursprüngliche, deutsche Fassung oder die Übersetzung der Fehlermeldung. Das Attribut "code" gibt die Nummer (Fehlercode) der Fehlermeldung an.

Abbildung II.3.8. Fehlermeldung



Dieser Typ ist eine Erweiterung des Basistyps `xs:string`.

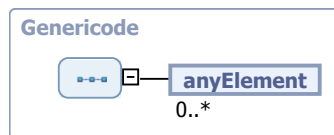
Kindelement von Fehlermeldung				
Kindelement	Typ	Anz.	Ref.	Seite
<code>code</code>	<code>xs:int</code>	1		
Nummer (Fehlercode) der Fehlermeldung				

II.3.1.9 Genericcode

Typ: `Genericcode`

Codeliste im Genericcode-Format.

Abbildung II.3.9. Genericcode

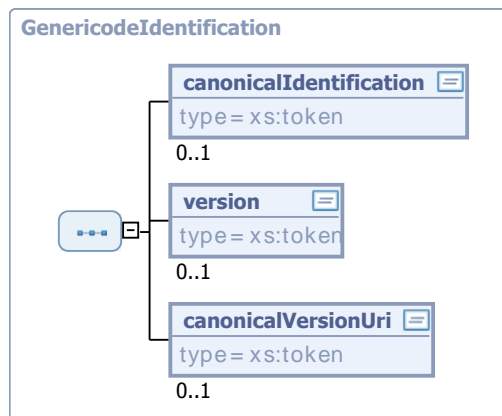


II.3.1.10 GenericcodeIdentification

Typ: `GenericcodeIdentification`

Identifikation einer Genericcode-Codeliste.

Abbildung II.3.10. GenericcodeIdentification



Kindelemente von GenericcodeIdentification				
Kindelement	Typ	Anz.	Ref.	Seite
<code>canonicalIdentification</code>	<code>xs:token</code>	0..1		
Kanonische ID der Genericcode-Codeliste.				
<code>version</code>	<code>xs:token</code>	0..1		
Version der Genericcode-Codeliste.				
<code>canonicalVersionUri</code>	<code>xs:token</code>	0..1		
Versions-URI der Genericcode-Codeliste.				

II.3.1.11 Header

Typ: **Header**

Technische Angaben für alle Nachrichten.

Abbildung II.3.11. Header



Kindelemente von Header				
Kindelement	Typ	Anz.	Ref.	Seite
nachrichtID	xs:token	1		
ID der Nachricht. Die ID kann vom Absender der Nachricht frei gewählt werden und wird in der Antwort als "referenzID" aufgenommen. Der Absender muss die ID so wählen, dass er die Antwort sicher zuordnen kann.				
erstellungzeitpunkt	xs:dateTime	1		
Datum, an dem die Nachricht erzeugt wurde.				
referenzID	xs:token	0..1		
NachrichtenID der Anfrage. Nur bei Antworten.				
produktbezeichnung	String.Latin	0..1	II.A.1	45
Der Name des Produkts, mit dem die Nachricht erstellt worden ist.				
produkthersteller	String.Latin	0..1	II.A.1	45
Der Name der Firma oder der Organisation, die für das Produkt verantwortlich ist, mit dem die Nachricht erstellt wurde.				

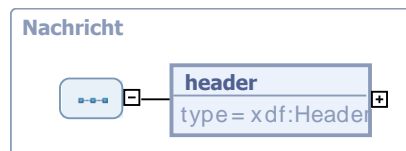
Kindelemente von Header				
Kindelement	Typ	Anz.	Ref.	Seite
produktversion	String.Latin	0..1	II.A.1	45
Die Version des Produkts, mit dem die Nachricht erstellt wurde (z.B. Versionsnummer mit Patchlevel).				
datenqualitaet	Code.Datenqualitaet	0..1	II.3.2.3	42
Die Qualität (Test- oder produktive Daten) der Inhalte der Nachricht.				

II.3.1.12 Nachricht

Typ: **Nachricht** (abstrakt)

Abstrakte Basisklasse für alle Nachrichten.

Abbildung II.3.12. Nachricht



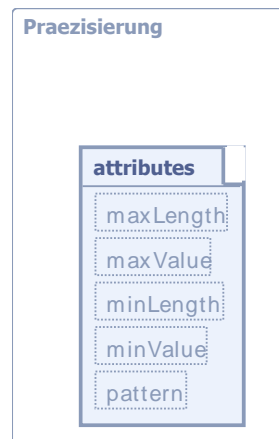
Kindelement von Nachricht				
Kindelement	Typ	Anz.	Ref.	Seite
header	Header	1	II.3.1.11	29
Technische Angaben für alle Nachrichten.				

II.3.1.13 Praezisierung

Typ: **Praezisierung**

Mit der Präzisierung kann der Datentyp eines Datenfeldes weiter eingeschränkt werden. Es sind Beschränkungen der Länge und des Wertebereichs möglich.

Abbildung II.3.13. Praezisierung



Dieser Typ ist eine Erweiterung des Basistyps `string.Latin` (siehe [Abschnitt II.A.1 auf Seite 45](#)).

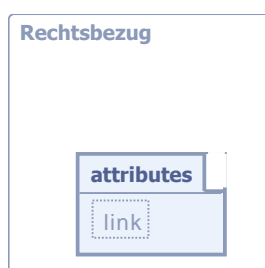
Kindelemente von Praezisierung				
Kindelement	Typ	Anz.	Ref.	Seite
<i>maxLength</i>	<code>xs:positiveInteger</code>	0..1		
Maximale Länge der einzugebenden Daten.				
<i>maxValue</i>	<code>xs:token</code>	0..1		
Maximaler Wert der angegeben werden muss. Dies ist nur bei geeigneten Datentypen wie Zahlen oder Datumsangaben möglich. Der angegebene Wert ist als Eingabe gültig (inklusive).				
<i>minLength</i>	<code>xs:nonNegativeInteger</code>	0..1		
Minimale Länge der einzugebenden Daten.				
<i>minValue</i>	<code>xs:token</code>	0..1		
Minimaler Wert der angegeben werden muss. Dies ist nur bei geeigneten Datentypen wie Zahlen oder Datumsangaben möglich. Der angegebene Wert ist als Eingabe gültig (inklusive).				
<i>pattern</i>	<code>xs:string</code>	0..1		
Regulärer Ausdruck zur Prüfung der Eingabe entsprechend dem HTML-Attribut "pattern". Siehe: https://developer.mozilla.org/en-US/docs/Web/HTML/Attributes/pattern				

II.3.1.14 Rechtsbezug

Typ: **Rechtsbezug**

Mit der Klasse "Rechtsbezug" werden strukturierte Rechtsbezüge möglich. Jeder Rechtsbezug wird als eigenes Element exportiert und kann einen Link auf die Rechtsnorm enthalten.

Abbildung II.3.14. Rechtsbezug



Dieser Typ ist eine Erweiterung des Basistyps `string.Latin` (siehe [Abschnitt II.A.1 auf Seite 45](#)).

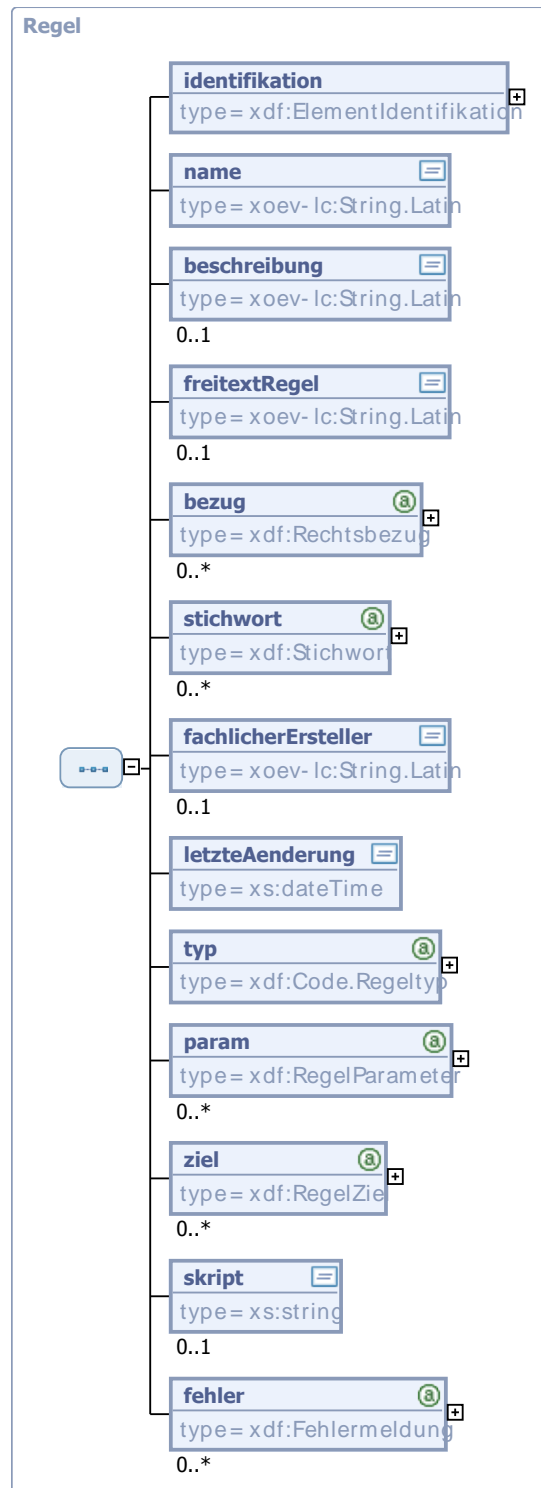
Kindelement von Rechtsbezug				
Kindelement	Typ	Anz.	Ref.	Seite
<i>link</i>	<code>xs:anyURI</code>	0..1		
Link auf den Gesetzestext im Internet.				

II.3.1.15 Regel

Typ: **Regel1**

Eine Regel enthält ein Skript, welches die technische Repräsentation einer zu erfüllenden Bedingung wiedergibt.

Abbildung II.3.15. Regel



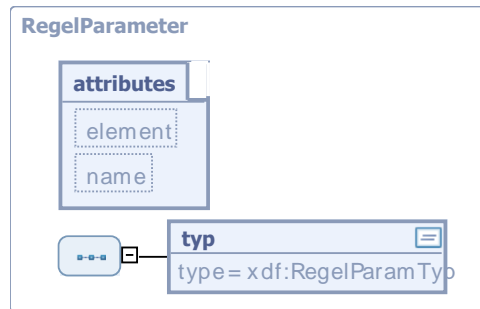
Kindelemente von Regel				
Kindelement	Typ	Anz.	Ref.	Seite
identifikation	ElementIdentifikation	1	II.3.1.5	24
Eine eindeutige Identifikation, die aus einer alphanumerischen und fortlaufenden Nummerierung bestehenden ID (Beispiel: "F00000000001") und der Versionsnummer zusammengesetzt ist.				
name	String.Latin	1	II.A.1	45
FIM-interner Name des Elements.				
beschreibung	String.Latin	0..1	II.A.1	45
Zusätzliche Beschreibung und Erläuterung des Elements für FIM- Nutzer.				
freitextRegel	String.Latin	0..1	II.A.1	45
Fasst den Inhalt einer Regel aus redaktioneller Sicht zusammen, d.h. aus Sicht der Rollen Ersteller, Methodenexperte, Informationsmanager, FIM- Koordinierungsstelle, FIM-Nutzer.				
bezug	Rechtsbezug	0..n	II.3.1.14	31
Bezug zu Rechtsnorm oder Standardisierungsvorhaben: Dieses Attribut ermöglicht es, Bezüge zu einschlägigen Rechtsnormen sowie zu in anderen Vorhaben wie den XÖV-Vorhaben (z.B. XWaffe) oder Vorhaben wie P23R gleich oder ähnlich definierten Datencontainern zu erstellen und diese Verbindung zu dokumentieren. Hierdurch lässt sich die Ausgestaltung eines Stammdatenschemas gemäß einschlägigen Rechtsnormen oder Standardisierungsverfahren belegen.				
stichwort	Stichwort	0..n	II.3.1.23	39
Stichwörter sind einerseits frei wählbare Begriffe, die den Elementen zugeordnet werden. Sie können jedoch auch eine Kategorie referenzieren und Werte aus dieser zuordnen.				
fachlicherErsteller	String.Latin	0..1	II.A.1	45
Ersteller oder erstellende Institution.				
letzteAenderung	xs:dateTime	1		
Datum der letzten Änderung. Wird vom FIM-System automatisch gepflegt.				
typ	Code.Regeltyp	1	II.3.2.9	43
Typ der Regel, z.B. Validierungsregel.				
param	RegelParameter	0..n	II.3.1.16	33
Eingabewert (Parameter) für das Script.				
ziel	RegelZiel	0..n	II.3.1.18	35
Element, auf das die Regel wirkt.				
skript	xs:string	0..1		
Das Skript, welches die technische Repräsentation einer zu erfüllenden Bedingung wiedergibt.				
fehler	Fehlermeldung	0..n	II.3.1.8	27
Fehlermeldung des Skripts.				

II.3.1.16 RegelParameter

Typ: **RegelParameter**

Eingabewert (Parameter) für das Script.

Abbildung II.3.16. RegelParameter



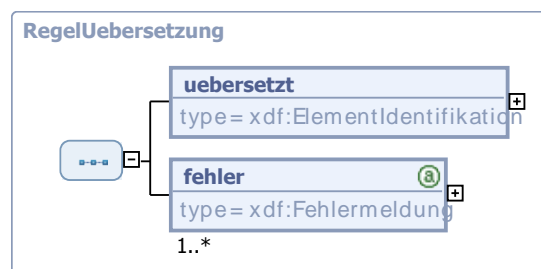
Kindelemente von RegelParameter				
Kindelement	Typ	Anz.	Ref.	Seite
element	xs:string	1		
Datenfeld, dessen Wert als Parameter an das Script übergeben wird.				
name	xs:string	1		
Name des Parameters, unter dem das Script auf den Wert zugreifen kann.				
typ	RegelParamTyp	1		
Typ des Auslösers; definiert den Inhalt des Parameters, der an das Skript übergeben wird.				

II.3.1.17 RegelUebersetzung

Typ: **RegelUebersetzung**

Übersetzung aller Fehlermeldungen einer Regel.

Abbildung II.3.17. RegelUebersetzung



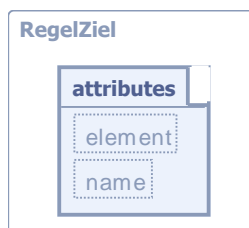
Kindelemente von RegelUebersetzung				
Kindelement	Typ	Anz.	Ref.	Seite
uebersetzt	ElementIdentifikation	1	II.3.1.5	24
Referenz auf die übersetzte Regel.				
fehler	Fehlermeldung	1..n	II.3.1.8	27
Übersetzung einer Fehlermeldung einer Regel. Das Element enthält die Übersetzung, das Attribut "code" gibt die Nummer (Fehlercode) der Fehlermeldung an.				

II.3.1.18 RegelZiel

Typ: **RegelZiel**

Element, auf das die Regel wirkt.

Abbildung II.3.18. RegelZiel



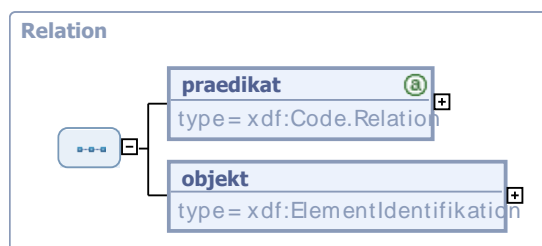
Kindelemente von RegelZiel				
Kindelement	Typ	Anz.	Ref.	Seite
element	xs:string	1		
Referenz auf das Element (Feld oder Gruppe), auf das die Regel wirkt.				
name	xs:string	1		
Name des Parameters, unter dem das Script auf die ID des Ziels zugreifen kann.				

II.3.1.19 Relation

Typ: **Relation**

Relationen dokumentieren Beziehungen der Elemente untereinander, wie z.B. "ist abgeleitet von", "ersetzt" oder "ist äquivalent zu".

Abbildung II.3.19. Relation



Kindelemente von Relation				
Kindelement	Typ	Anz.	Ref.	Seite
praedikat	Code.Relation	1	II.3.2.11	43
Art der Relationen, z.B. "ist abgeleitet von", "ersetzt" oder "ist äquivalent zu".				
objekt	ElementIdentifikation	1	II.3.1.5	24

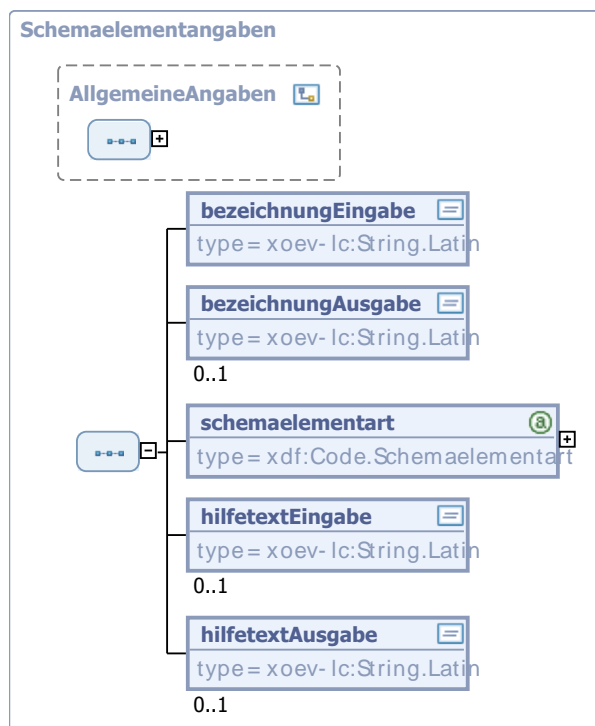
Kindelemente von Relation				
Kindelement	Typ	Anz.	Ref.	Seite
Element zu dem die Relationen besteht.				

II.3.1.20 Schemaelementangaben

Typ: `schemaelementangaben` (abstrakt)

Abstrakte Basisklasse für Elemente eines Stammdatenschemas (Gruppen und Felder).

Abbildung II.3.20. Schemaelementangaben



Dieser Typ ist eine Erweiterung des Basistyps `AllgemeineAngaben` (siehe [Abschnitt II.3.1.1 auf Seite 17](#)).

Kindelemente von Schemaelementangaben				
Kindelement	Typ	Anz.	Ref.	Seite
<code>bezeichnungEingabe</code>	<code>String.Latin</code>	1	II.A.1	45
Bezeichnung des Elements, welches für den Bürger/das Unternehmen auf Input-Formularen sichtbar ist.				
<code>bezeichnungAusgabe</code>	<code>String.Latin</code>	0..1	II.A.1	45
Bezeichnung des Elements, welches für den Bürger/das Unternehmen auf Output-Formularen sichtbar ist.				
<code>schemaelementart</code>	<code>Code.Schemaelementart</code>	1	II.3.2.12	43
Art des Schemaelements				
<code>hilfertextEingabe</code>	<code>String.Latin</code>	0..1	II.A.1	45
Erläuternder Hilfertext für den Formularanwender (Bürger/Unternehmen) eines Input-Dokuments, um zu verdeutlichen, welche Inhalte in einer Datenfeldgruppe einzugeben sind bzw. welche Aktion vorzunehmen ist.				

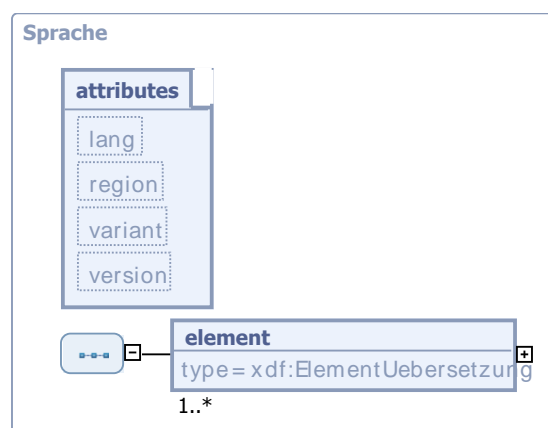
Kindelemente von Schemaelementangaben				
Kindelement	Typ	Anz.	Ref.	Seite
hilfetextAusgabe	String.Latin	0..1	II.A.1	45
Erläuternder Hilfetext für den Formularanwender (Bürger/Unternehmen) eines Output-Dokuments.				

II.3.1.21 Sprache

Typ: **sprache**

Übersetzung eines Stammdatenschemas in eine Sprache

Abbildung II.3.21. Sprache



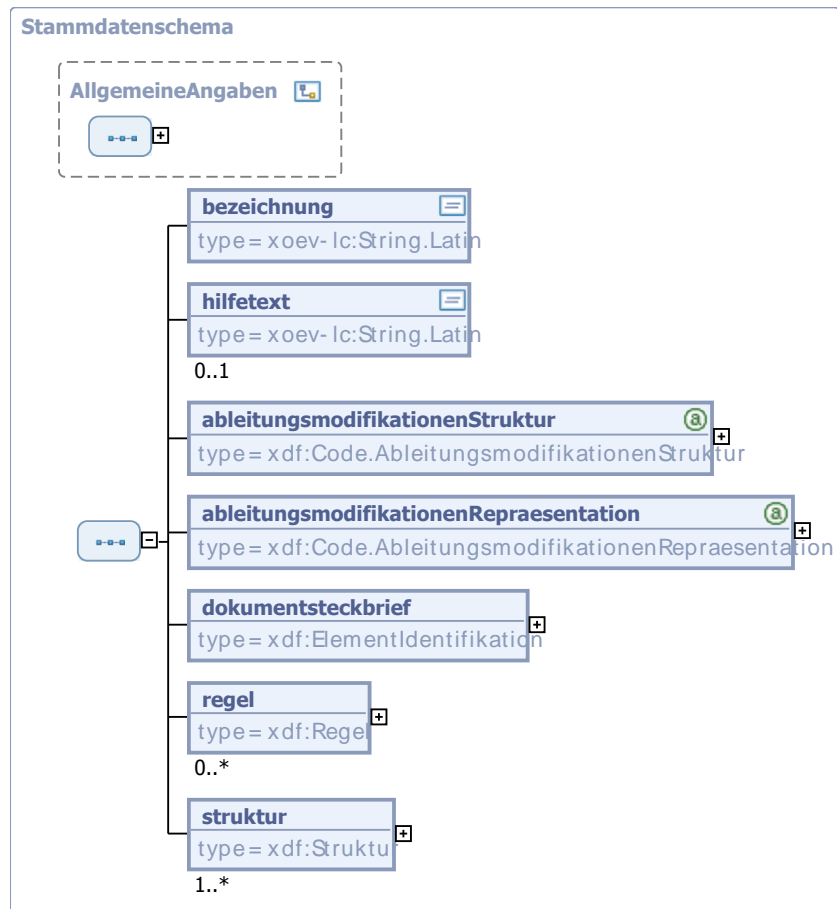
Kindelemente von <i>sprache</i>				
Kindelement	Typ	Anz.	Ref.	Seite
<i>lang</i>	Language	1	II.2.3	15
Die Primary Language darf nur aus zwei oder drei Kleinbuchstaben bestehen (gemäß ISO 639-1 (2002), ISO 639-2 (1998), ISO 639-3 (2007) oder ISO 639-5 (2008)).				
<i>region</i>	CountryCode	0..1	II.2.2	15
Ländercode, bestehend aus zwei Großbuchstaben gemäß ISO 3166-1 alpha-2. Beispiel: "DE" für Deutschland				
<i>variant</i>	LanguageVariant	0..1	II.2.4	15
Sprachvariante (gemäß IANA Registry), derzeit wird nur "simple" für einfache Sprache unterstützt.				
<i>version</i>	xs:positiveInteger	1		
Version der Übersetzung. Diese wird fortlaufend durchnummeriert.				
<i>element</i>	ElementUebersetzung	1..n	II.3.1.6	25
Referenz auf das übersetzte Stammdatenschema.				

II.3.1.22 Stammdatenschema

Typ: **stammdatenschema**

Die FIM-Stammdatenschemata werden auf allen föderalen Ebenen auf Basis von Gesetzen, Verordnungen sowie Verwaltungs- und Ausführungsvorschriften unter Anwendung des Redaktionsprozesses entwickelt. Stammdatenschemata sollen Allgemeingültigkeit besitzen.

Abbildung II.3.22. Stammdatenschema



Dieser Typ ist eine Erweiterung des Basistyps **AllgemeineAngaben** (siehe [Abschnitt II.3.1.1 auf Seite 17](#)).

Kindelemente von Stammdatenschema				
Kindelement	Typ	Anz.	Ref.	Seite
bezeichnung	<code>String.Latin</code>	1	II.A.1	45
Bezeichnung des Stammdatenschemas, welches für den Bürger/das Unternehmen auf Input-Formularen sichtbar ist.				
hilfetext	<code>String.Latin</code>	0..1	II.A.1	45
Hilfetext zu diesem Element.				
ableitungsmodifikationenStruktur	<code>Code.AbleitungsmodifikationenStruktur</code>	1	II.3.2.2	42
Strukturelle Ableitungsmodifikation.				
ableitungsmodifikationenRepraesentation	<code>Code.AbleitungsmodifikationenRepraesentation</code>	1	II.3.2.1	42
Ableitungsmodifikation der Repräsentation.				
dokumentsteckbrief	<code>ElementIdentifikation</code>	1	II.3.1.5	24
Referenz auf den Dokumentsteckbrief, dem dieses Stammdatenschema zugeordnet ist.				

Kindelemente von Stammdatenschema				
Kindelement	Typ	Anz.	Ref.	Seite
regel	Regel	0..n	II.3.1.15	31
Die Regeln geben Bedingungen für korrekt befüllte Stammdatenschemata vor.				
struktur	Struktur	1..n	II.3.1.24	39
Die Struktur listet alle Unterelemente auf.				

II.3.1.23 Stichwort

Typ: **stichwort**

Stichwörter sind frei wählbare Begriffe, die den Elementen zugeordnet werden. Sie können jedoch auch eine Kategorie referenzieren und Werte aus dieser zuordnen.

Abbildung II.3.23. Stichwort



Dieser Typ ist eine Erweiterung des Basistyps `xs:token`.

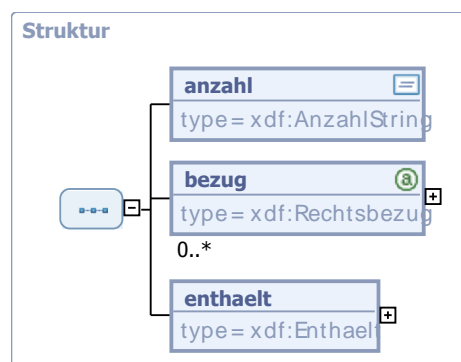
Kindelement von stichwort				
Kindelement	Typ	Anz.	Ref.	Seite
uri	<code>xs:anyURI</code>	1		
URI der Stichwort-Kategorie. Diese kann eine Codeliste referenzieren. In diesem Fall sind nur die Einträge der Codeliste als Stichwörter zugelassen.				

II.3.1.24 Struktur

Typ: **struktur**

Die Struktur listet ein Unterelement eines übergeordneten Elements auf.

Abbildung II.3.24. Struktur



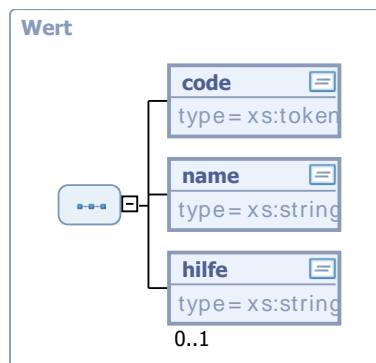
Kindelemente von Struktur				
Kindelement	Typ	Anz.	Ref.	Seite
anzahl	AnzahlString	1	II.2.1	15
Die Anzahl definiert, wie oft das Unterelement minimal und maximal enthalten sein kann.				
bezug	Rechtsbezug	0..n	II.3.1.14	31
Bezug zu Rechtsnorm oder Standardisierungsvorhaben: Dieses Attribut ermöglicht es, Bezüge zu einschlägigen Rechtsnormen sowie zu in anderen Vorhaben wie den XÖV-Vorhaben (z.B. XWaffe) oder Vorhaben wie P23R gleich oder ähnlich definierten Datencontainern zu erstellen und diese Verbindung zu dokumentieren. Hierdurch lässt sich die Ausgestaltung eines Stammdatenschemas gemäß einschlägigen Rechtsnormen oder Standardisierungsverfahren belegen.				
enthalt	Enthalt	1	II.3.1.7	27
Das enthaltene Unterelement.				

II.3.1.25 Wert

Typ: **wert**

Ein Wert einer Werteliste. Vergleichbar mit einer Zeile einer Codeliste.

Abbildung II.3.25. Wert



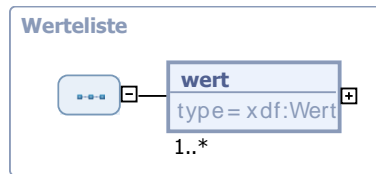
Kindelemente von Wert				
Kindelement	Typ	Anz.	Ref.	Seite
code	xs:token	1		
Code des Werts. Dieser ist eindeutig in der Werteliste.				
name	xs:string	1		
Name des Werts. Dieser ist das an der Oberfläche anzuzeigende.				
hilfe	xs:string	0..1		
Hilfe oder Erläuterung zu diesem Wert.				

II.3.1.26 Werteliste

Typ: **werteliste**

Eine Liste von Werten, die ein Datenfeld annehmen kann.

Abbildung II.3.26. Werteliste



Kindelement von werteliste				
Kindelement	Typ	Anz.	Ref.	Seite
wert	Wert	1..n	II.3.1.25	40
Ein Wert der Werteliste. Vergleichbar mit einer Zeile einer Codeliste.				

II.3.2 Codes und Codelisten

Name	# Einträge	Einträge	Code-Datentyp
AbleitungsmodifikationenRepraesentation	2	Seite 59	Seite 42
AbleitungsmodifikationenStruktur	4	Seite 60	Seite 42
Art (Typ) der Regel	4	Seite 61	Seite 43
Art (Typ) des Auslösers der Regel	4	Seite 62	Seite 43
Art der Gruppe	1	Seite 63	Seite 43
Datenfeldart	5	Seite 64	Seite 42
Datenfeldtyp	11	Seite 65	Seite 42
Datenqualität	2	Seite 66	Seite 42
Schemaelementart	3	Seite 67	Seite 43
Vorbefüllung von Datenfeldern	3	Seite 68	Seite 43

II.3.2.1 Code.AbleitungsmodifikationenRepraesentation

Code	Code.AbleitungsmodifikationenRepraesentation
Codelisten-Nutzung	Typ: 1, Inhalte der Codeliste siehe Seite 59
Codelisten-URI	urn:xoev-de:fim:codeliste:xdatenfelder.ableitungsmodifikationenRepraesentation
Codelisten-Version	1.0

II.3.2.2 Code.AbleitungsmodifikationenStruktur

Code	Code.AbleitungsmodifikationenStruktur
Codelisten-Nutzung	Typ: 1, Inhalte der Codeliste siehe Seite 60
Codelisten-URI	urn:xoev-de:fim:codeliste:xdatenfelder.ableitungsmodifikationenStruktur
Codelisten-Version	1.0

II.3.2.3 Code.Datenqualitaet

Code	Code.Datenqualitaet
Codelisten-Nutzung	Typ: 1, Inhalte der Codeliste siehe Seite 66
Codelisten-URI	urn:xoev-de:fim:codeliste:datenfelder.datenqualitaet
Codelisten-Version	1.0

II.3.2.4 Code.Datentyp

Code	Code.Datentyp
Codelisten-Nutzung	Typ: 1, Inhalte der Codeliste siehe Seite 65
Codelisten-URI	urn:xoev-de:fim:codeliste:xdatenfelder.datentyp
Codelisten-Version	2.0

II.3.2.5 Code.Dokumentart

Code	Code.Dokumentart
Codelisten-Nutzung	Typ: 3, siehe Beschreibung
Codelisten-URI	urn:xoev-de:fim-datenfelder:codeliste:dokumentart
Codelisten-Version	unbestimmt

II.3.2.6 Code.Feldart

Code	Code.Feldart
Codelisten-Nutzung	Typ: 1, Inhalte der Codeliste siehe Seite 64
Codelisten-URI	urn:xoev-de:fim:codeliste:xdatenfelder.feldart
Codelisten-Version	2.0

II.3.2.7 Code.Freigabestatus

Code	Code.Freigabestatus
Codelisten-Nutzung	Typ: 3, siehe Beschreibung

Code	Code.Freigabestatus
Codelisten-URI	urn:xoev-de:xprozess:codeliste:status
Codelisten-Version	unbestimmt

II.3.2.8 Code.Gruppenart

Code	Code.Gruppenart
Codelisten-Nutzung	Typ: 1, Inhalte der Codeliste siehe Seite 63
Codelisten-URI	urn:xoev-de:fim:codeliste:datenfelder.gruppenart
Codelisten-Version	1.0

II.3.2.9 Code.Regeltyp

Code	Code.Regeltyp
Codelisten-Nutzung	Typ: 1, Inhalte der Codeliste siehe Seite 61
Codelisten-URI	urn:xoev-de:fim:codeliste:xdatenfelder.regeltyp
Codelisten-Version	1.0

II.3.2.10 Code.RegelParamTyp

Code	Code.RegelParamTyp
Codelisten-Nutzung	Typ: 1, Inhalte der Codeliste siehe Seite 62
Codelisten-URI	urn:xoev-de:fim:codeliste:xdatenfelder.regelparamtyp
Codelisten-Version	1.0

II.3.2.11 Code.Relation

Code	Code.Relation
Codelisten-Nutzung	Typ: 3, siehe Beschreibung
Codelisten-URI	urn:xoev-de:fim-datenfelder:codeliste:relation
Codelisten-Version	unbestimmt

II.3.2.12 Code.Schemaelementart

Code	Code.Schemaelementart
Codelisten-Nutzung	Typ: 1, Inhalte der Codeliste siehe Seite 67
Codelisten-URI	urn:xoev-de:fim:codeliste:xdatenfelder.schemaelementart
Codelisten-Version	1.0

II.3.2.13 Code.Vorbefuellung

Code	Code.Vorbefuellung
Codelisten-Nutzung	Typ: 1, Inhalte der Codeliste siehe Seite 68
Codelisten-URI	urn:xoev-de:fim:codeliste:xdatenfelder.vorbefuellung

Code	Code.Vorbefuellung
Codelisten-Version	1.0

II.A Eingebundene externe Modelle



Folgende externe Modelle werden in dieser Spezifikation verwendet und sind auf den XÖV-Webseiten (siehe <http://www.xoev.de/de/produkte>) oder im XRepository (siehe <http://www.xrepository.de>) veröffentlicht:

II.A.1 XOEV-Bibliothek

XOEV-Bibliothek; Fassung 2017-08-01

Folgende Datentypen aus dem externen Modell werden in dieser Spezifikation verwendet:

- Code
- String.Latin



III Datenübermittlung

III.1 Übertragung von Inhalten **XÖV**

Übertragung von Inhalten des Bausteins Datenfelder

III.1.1 Die Nachrichten

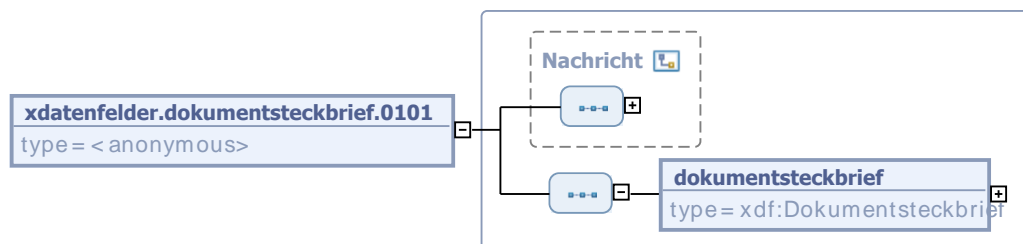
Bezeichnung	Nr.	Beschreibung	Seite
Schema-Datei: xdatenfelder-baukasten.xsd			
Schema-Datei: xdatenfelder-datentypen.xsd			
Schema-Datei: xdatenfelder.xsd			
xdatenfelder.dokumentsteckbrief.0101	0101	Dokumentsteckbrief übertragen	Seite 49
xdatenfelder.stammdatenschema.0102	0102	Stammdatenschema übertragen	Seite 50
xdatenfelder.datenfeldgruppe.0103	0103	Datenfeldgruppe übertragen	Seite 50
xdatenfelder.datenfeld.0104	0104	Datenfeld übertragen	Seite 51
xdatenfelder.codeliste.0105	0105	Codeliste übertragen	Seite 51
xdatenfelder.uebersetzung.0106	0106	Übersetzung(en) eines Stammdatenschemas übertragen	Seite 52
Schema-Datei: xdatenfelder_verzeichnis.xsd			
verzeichnis.datenfeldbaukasten.0201	0201	Datenfeldbaukasten übertragen.	Seite 52
verzeichnis.datenfeldkatalog.0202	0202	Datenfeldkatalog übertragen.	Seite 53
verzeichnis.stammdatenschemabibliothek.0203	0203	Stammdatenschemabibliothek übertragen.	Seite 53

III.1.1.1 xdatenfelder.dokumentsteckbrief.0101

Nachricht: `xdatenfelder.dokumentsteckbrief.0101`

Dokumentsteckbrief übertragen

Abbildung III.1.1. xdatenfelder.dokumentsteckbrief.0101



Diese Nachricht ist eine Erweiterung des Basistyps **Nachricht** (siehe [Abschnitt II.3.1.12 auf Seite 30](#)).

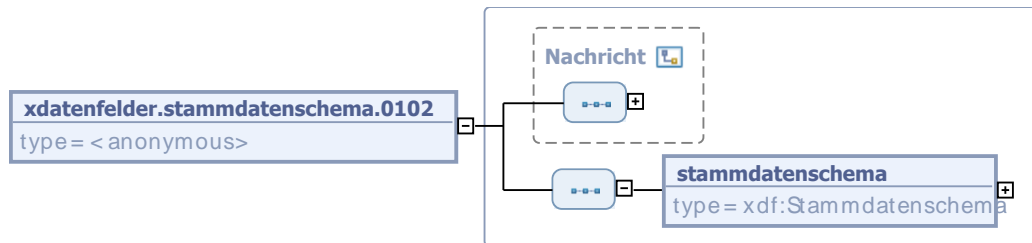
Kindelement von <code>xdatenfelder.dokumentsteckbrief.0101</code>				
Kindelement	Typ	Anz.	Ref.	Seite
<code>dokumentsteckbrief</code>	Dokumentsteckbrief	1	II.3.1.4	23
Der Dokumentsteckbrief.				

III.1.1.2 `xdatenfelder.stammdatenschema.0102`

Nachricht: `xdatenfelder.stammdatenschema.0102`

Stammdatenschema übertragen

Abbildung III.1.2. `xdatenfelder.stammdatenschema.0102`



Diese Nachricht ist eine Erweiterung des Basistyps **Nachricht** (siehe [Abschnitt II.3.1.12 auf Seite 30](#)).

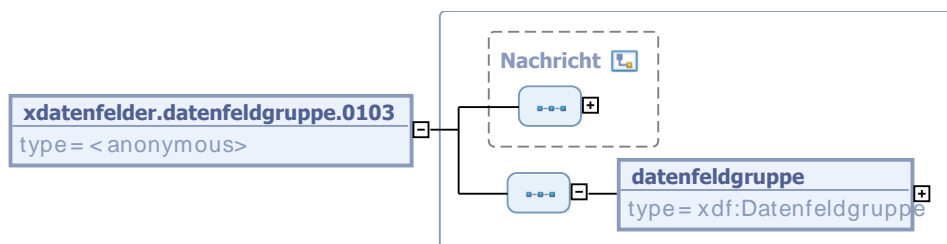
Kindelement von <code>xdatenfelder.stammdatenschema.0102</code>				
Kindelement	Typ	Anz.	Ref.	Seite
<code>stammdatenschema</code>	Stammdatenschema	1	II.3.1.22	37
Das Stammdatenschema.				

III.1.1.3 `xdatenfelder.datenfeldgruppe.0103`

Nachricht: `xdatenfelder.datenfeldgruppe.0103`

Datenfeldgruppe übertragen

Abbildung III.1.3. `xdatenfelder.datenfeldgruppe.0103`



Diese Nachricht ist eine Erweiterung des Basistyps **Nachricht** (siehe [Abschnitt II.3.1.12 auf Seite 30](#)).

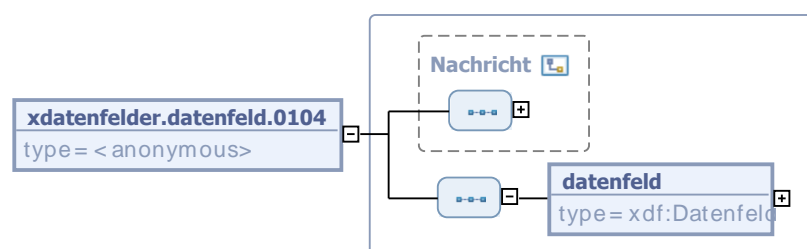
Kindelement von <code>xdatenfelder.datenfeldgruppe.0103</code>				
Kindelement	Typ	Anz.	Ref.	Seite
<code>datenfeldgruppe</code>	Datenfeldgruppe	1	II.3.1.3	22
Die Datenfeldgruppe.				

III.1.1.4 `xdatenfelder.datenfeld.0104`

Nachricht: `xdatenfelder.datenfeld.0104`

Datenfeld übertragen

Abbildung III.1.4. `xdatenfelder.datenfeld.0104`



Diese Nachricht ist eine Erweiterung des Basistyps **Nachricht** (siehe [Abschnitt II.3.1.12 auf Seite 30](#)).

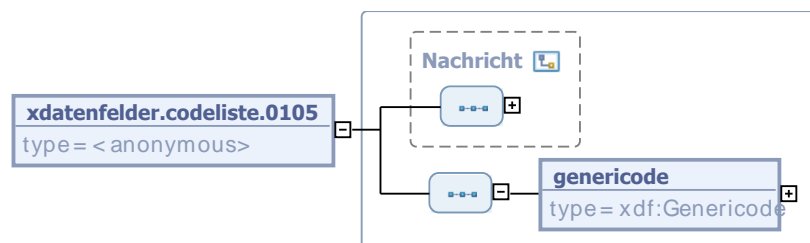
Kindelement von <code>xdatenfelder.datenfeld.0104</code>				
Kindelement	Typ	Anz.	Ref.	Seite
<code>datenfeld</code>	Datenfeld	1	II.3.1.2	20
Das Datenfeld.				

III.1.1.5 `xdatenfelder.codeliste.0105`

Nachricht: `xdatenfelder.codeliste.0105`

Codeliste übertragen

Abbildung III.1.5. `xdatenfelder.codeliste.0105`



Diese Nachricht ist eine Erweiterung des Basistyps **Nachricht** (siehe [Abschnitt II.3.1.12 auf Seite 30](#)).

Kindelement von <code>xdatenfelder.codeliste.0105</code>				
Kindelement	Typ	Anz.	Ref.	Seite
<code>genericcode</code>	Genericcode	1	II.3.1.9	28

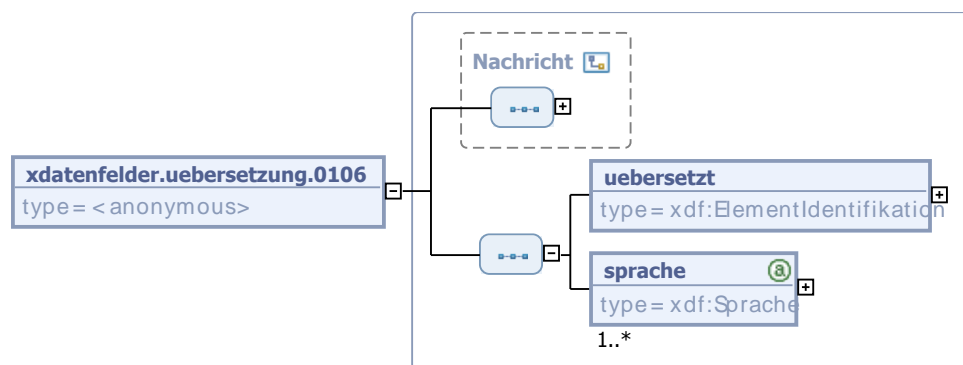
Kindelement von <code>xdatenfelder.codeliste.0105</code>				
Kindelement	Typ	Anz.	Ref.	Seite
Die Codeliste im Genericcode Format				

III.1.1.6 `xdatenfelder.uebersetzung.0106`

Nachricht: `xdatenfelder.uebersetzung.0106`

Übersetzung(en) eines Stammdatenschemas übertragen

Abbildung III.1.6. `xdatenfelder.uebersetzung.0106`



Diese Nachricht ist eine Erweiterung des Basistyps `Nachricht` (siehe [Abschnitt II.3.1.12 auf Seite 30](#)).

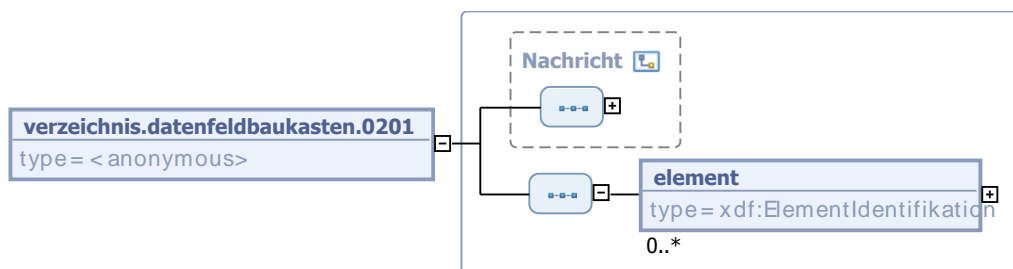
Kindelemente von <code>xdatenfelder.uebersetzung.0106</code>				
Kindelement	Typ	Anz.	Ref.	Seite
<code>uebersetzt</code>	<code>ElementIdentifikation</code>	1	II.3.1.5	24
Referenz auf das übersetzte Stammdatenschema				
<code>sprache</code>	<code>Sprache</code>	1..n	II.3.1.21	37
Es können mehrere Übersetzungen in einer Datei übertragen werden. Hierzu wird das Element "sprache" pro Übersetzung angegeben.				

III.1.1.7 `verzeichnis.datenfeldbaukasten.0201`

Nachricht: `verzeichnis.datenfeldbaukasten.0201`

Datenfeldbaukasten übertragen.

Abbildung III.1.7. `verzeichnis.datenfeldbaukasten.0201`



Diese Nachricht ist eine Erweiterung des Basistyps **Nachricht** (siehe [Abschnitt II.3.1.12 auf Seite 30](#)).

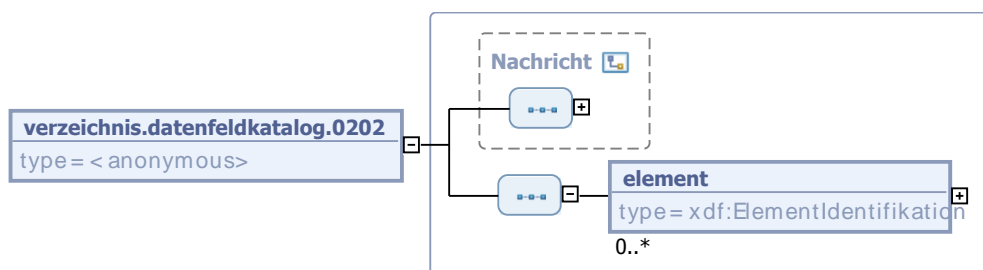
Kindelement von <code>verzeichnis.datenfeldbaukasten.0201</code>				
Kindelement	Typ	Anz.	Ref.	Seite
element	ElementIdentifikation	0..n	II.3.1.5	24
Liste der IDs aller veröffentlichten Felder, Gruppen und Codelisten.				

III.1.1.8 `verzeichnis.datenfeldkatalog.0202`

Nachricht: `verzeichnis.datenfeldkatalog.0202`

Datenfeldkatalog übertragen.

Abbildung III.1.8. `verzeichnis.datenfeldkatalog.0202`



Diese Nachricht ist eine Erweiterung des Basistyps **Nachricht** (siehe [Abschnitt II.3.1.12 auf Seite 30](#)).

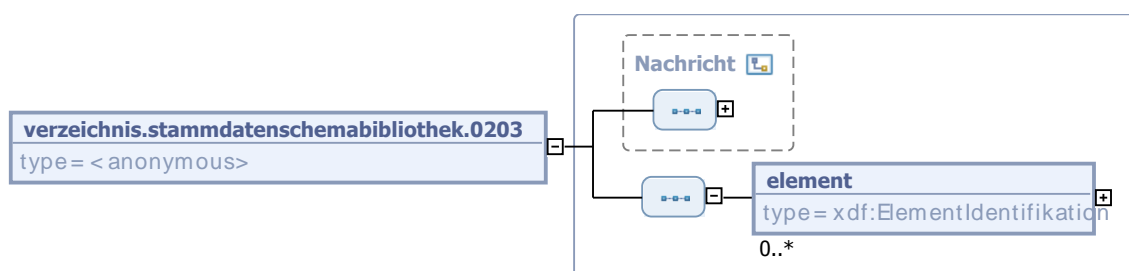
Kindelement von <code>verzeichnis.datenfeldkatalog.0202</code>				
Kindelement	Typ	Anz.	Ref.	Seite
element	ElementIdentifikation	0..n	II.3.1.5	24
Liste der IDs aller veröffentlichten Dokumentsteckbriefe.				

III.1.1.9 `verzeichnis.stammdatenschemabibliothek.0203`

Nachricht: `verzeichnis.stammdatenschemabibliothek.0203`

Stammdatenschemabibliothek übertragen.

Abbildung III.1.9. `verzeichnis.stammdatenschemabibliothek.0203`



Diese Nachricht ist eine Erweiterung des Basistyps **Nachricht** (siehe [Abschnitt II.3.1.12 auf Seite 30](#)).

Kindelement von <code>verzeichnis.stammdatenschemabibliothek.0203</code>				
Kindelement	Typ	Anz.	Ref.	Seite
element	ElementIdentifikation	0..n	II.3.1.5	24
Liste der IDs aller veröffentlichten Stammdatenschemata.				



IV Anhänge

IV.A Übersicht über alle Nachrichten



Bezeichnung	Nr.	Beschreibung	Seite
Schema-Datei: xdatenfelder-baukasten.xsd			
Schema-Datei: xdatenfelder-datentypen.xsd			
Schema-Datei: xdatenfelder.xsd			
xdatenfelder.dokumentsteckbrief.0101	0101	Dokumentsteckbrief übertragen	Seite 49
xdatenfelder.stammdatenschema.0102	0102	Stammdatenschema übertragen	Seite 50
xdatenfelder.datenfeldgruppe.0103	0103	Datenfeldgruppe übertragen	Seite 50
xdatenfelder.datenfeld.0104	0104	Datenfeld übertragen	Seite 51
xdatenfelder.codeliste.0105	0105	Codeliste übertragen	Seite 51
xdatenfelder.uebersetzung.0106	0106	Übersetzung(en) eines Stammdatenschemas übertragen	Seite 52
Schema-Datei: xdatenfelder_verzeichnis.xsd			
verzeichnis.datenfeldbaukasten.0201	0201	Datenfeldbaukasten übertragen.	Seite 52
verzeichnis.datenfeldkatalog.0202	0202	Datenfeldkatalog übertragen.	Seite 53
verzeichnis.stammdatenschemabibliothek.0203	0203	Stammdatenschemabibliothek übertragen.	Seite 53

IV.B Die Codelisten des Standards XDatenfelder



IV.B.1 Details

IV.B.1.1 Codeliste AbleitungsmodifikationenRepraesentation

Codeliste	AbleitungsmodifikationenRepraesentation (XDatenfelder.AbleitungsmodifikationenRepraesentation)
Herausgeber	FIM Baustein Datenfelder
Beschreibung	Mit dieser Codeliste wird die Möglichkeit der Modifikation eines abgeleiteten Elements beschrieben.
Code (code)	Beschreibung (bezeichnung)
0	nicht modifizierbar
1	modifizierbar

IV.B.1.2 Codeliste AbleitungsmodifikationenStruktur

Codeliste	AbleitungsmodifikationenStruktur (XDatenfelder.AbleitungsmodifikationenStruktur)
Herausgeber	FIM Baustein Datenfelder
Beschreibung	Mit dieser Codeliste wird die Möglichkeit der Veränderungen an der Struktur von abgeleiteten Elementen beschrieben.
Code (code)	Beschreibung (bezeichnung)
3	alles modifizierbar
2	nur erweiterbar
1	nur einschränkbar
0	nicht modifizierbar

IV.B.1.3 Codeliste Art (Typ) der Regel

Codeliste	Art (Typ) der Regel (Regeltyp)
Herausgeber	FIM Baustein Datenfelder
Beschreibung	Diese Codeliste definiert die verschiedenen Arten von Regeln.
Code (code)	Beschreibung (bezeichnung)
K	Komplexe Regel
M	Multiplizitätsregel
V	Validierungsregel
B	Berechnungsregel

IV.B.1.4 Codeliste Art (Typ) des Auslösers der Regel

Codeliste	Art (Typ) des Auslösers der Regel (RegelParamTyp)
Herausgeber	FIM Baustein Datenfelder
Beschreibung	Diese Codeliste definiert, welchen Wert der Auslöser an das Skript der Regel übergibt.
Code (code)	Beschreibung (beschreibung)
wert	Übergibt den Wert des Elements.
anzahl	Übergibt die Anzahl des Elements.
selektor	Übergibt einen Selektor auf das Element.
objekt	Übergibt eine Datenstruktur, die diverse Angaben zum Element enthält.

IV.B.1.5 Codeliste Art der Gruppe

Codeliste	Art der Gruppe (XDatenfelder.Gruppenart)
Herausgeber	FIM Baustein Datenfelder
Beschreibung	Die Werte dieser Codeliste kennzeichnen Datenfeldgruppen mit spezieller Semantik.
Code (code)	Beschreibung (bezeichnung)
X	Auswahlgruppe

IV.B.1.6 Codeliste Datenfeldart

Codeliste	Datenfeldart (XDatenfelder.Feldart)
Herausgeber	FIM Baustein Datenfelder
Beschreibung	Mit dieser Codeliste wird die Feldart eines Datenfeldes abgebildet.
Code (code)	Beschreibung (bezeichnung)
input	Eingabe
select	Auswahl
label	statisch
hidden	versteckt
locked	gesperrt

IV.B.1.7 Codeliste Datenfeldtyp

Codeliste	Datenfeldtyp (XDatenfelder.Datentyp)
Herausgeber	FIM Baustein Datenfelder
Beschreibung	Mit dieser Codeliste wird der Datentyp eines Datenfeldes abgebildet.
Code (code)	Beschreibung (bezeichnung)
text	Text
text_latin	String.Latin+ 1.2 (DIN Spec 91379) Datentyp C
date	Datum
time	Zeit (Stunde und Minute)
datetime	Zeitpunkt (Datum und Zeit)
bool	Wahrheitswert
num	Nummer
num_int	Ganzzahl
num_currency	Geldbetrag
file	Anlage (Datei)
obj	Objekt (Blob)

IV.B.1.8 Codeliste Datenqualität

Codeliste	Datenqualität (XDatenfelder.Datenqualitaet)
Herausgeber	FIM Baustein Datenfelder
Beschreibung	Mit dieser Codeliste wird die Qualität (Test- oder produktive Daten) der Inhalte der Nachricht festgelegt.
Code (code)	Beschreibung (bezeichnung)
PROD	Produktive Daten
TEST	Testdaten

IV.B.1.9 Codeliste Schemaelementart

Codeliste	Schemaelementart (XDatenfelder.Schemaelementart)
Herausgeber	FIM Baustein Datenfelder
Beschreibung	Mit dieser Codeliste wird die Art eines Schemaelements abgebildet.
Code (code)	Beschreibung (bezeichnung)
ABS	abstrakt
HAR	harmonisiert
RNG	rechtsnormgebunden

IV.B.1.10 Codeliste Vorbefüllung von Datenfeldern

Codeliste	Vorbefüllung von Datenfeldern (Vorbefuellung)
Herausgeber	FIM Baustein Datenfelder
Beschreibung	Die Werte dieser Codeliste definieren, ob das Datenfeld von der Behörde vorbefüllt werden kann oder muss.
Code (code)	Beschreibung (bezeichnung)
keine	keine Vorbefüllung
optional	optionale Vorbefüllung
verpflichtend	verpflichtende Vorbefüllung

IV.C Mitwirkende



Tabelle IV.C.1. Versionshistorie

Name	Institution
Gettkandt, Jessica	FJD Information Technologies AG
Huber, Andreas	FJD Information Technologies AG
Jorga, Frank	FJD Information Technologies AG
Maid, Thorsten	FITKO (Föderale IT-Kooperation), Produktmanagement FIM
Nax, Kirsten	Ministerium für Inneres und Sport Niedersachsen, IT-Services
Schmitz, Volker	Volker Schmitz IT-Beratung

IV.D Versionshistorie



Tabelle IV.D.1. Versionshistorie

Datum	Version	Bearbeiter	Änderung
2017-12-20		Andreas Huber, FJD	Dokument erstellt
2018-02-02	0.1	Andreas Huber, FJD	Vorbereitung für erste Veröffentlichung
2018-02-28		Andreas Huber, FJD	Kleinere Änderungen
2018-04-13	0.2	Andreas Huber, FJD	Umbenennung des Bausteins "Formulare" zu "Datenfelder"
2018-07-25	0.3	Andreas Huber, FJD	Regel um das Feld "script" ergänzt
2018-09-05	0.4	Andreas Aschauer, FJD Andreas Huber, FJD Frank Jorga, FJD	Nach XÖV modelliert
2018-09-05	0.5	Andreas Aschauer, FJD Andreas Huber, FJD	<ul style="list-style-type: none"> • Fehlerkorrekturen • Struktur: Element "bezug" hinzugefügt • Status als Codeliste modelliert • Allgemeine Angaben: Element "fachlicherErsteller" hinzugefügt
2018-10-10	1.0	Andreas Huber, FJD	<ul style="list-style-type: none"> • Allgemeine Angaben: Element "bezeichnung" in "bezeichnungEingabe" umbenannt • Allgemeine Angaben: Element "bezeichnungAusgabe" hinzugefügt • Dokumentensteckbrief: Element "hilfetext" hinzugefügt • Stammdatenschema: Element "hilfetext" hinzugefügt • Stammdatenschema: Element "ableitungsmodifikationen-Struktur" inkl. Codeliste hinzugefügt • Stammdatenschema: Element "ableitungsmodifikationen-Repraesentation" inkl. Codeliste hinzugefügt
2018-12-06	2.0	Andreas Huber, FJD	<ul style="list-style-type: none"> • Die nicht notwendige Version wurde aus den Nachrichtennummern entfernt. Sie bestehen jetzt nur noch aus einer Gruppen- und Nachrichtennummer, jeweils zweistellig. • Nachricht hinzugefügt: xdatenfelder.codeliste.0105 • Nachricht hinzugefügt: verzeichnis.datenfeldbaukasten.0201 • Nachricht hinzugefügt: verzeichnis.datenfeldkatalog.0202 • Nachricht hinzugefügt: verzeichnis.stammdatenschemabibliothek.0203 • Nachricht hinzugefügt: sync.anfrage.0301 • Nachricht hinzugefügt: sync.antwort.0302 • Nachricht hinzugefügt: sync.abruf.0303 • Nachricht hinzugefügt: such.anfrage.0401 • Nachricht hinzugefügt: such.antwort.0402

Datum	Version	Bearbeiter	Änderung
			<ul style="list-style-type: none">• Header: "nachrichtID" ergänzt• Header: "datum" geändert in "erstellungzeitpunkt" (dateTime)• Header: "referenzID" ergänzt• Header: "fimId" und "version" entfernt• Klasse "Codelist" erweitern für eine vollständige Codelist, inkl. Tabelle der Werte• Datenfeld: Das Element "codelist" in "codelistReferenz" umbenannt und auf eine Referenz reduziert• FimID: Die "id" und "version" in Unterelemente getrennt
2021-04-15	3.0	Andreas Huber, FJD	Mehrere größere Änderungen, siehe separates Dokument

Stichwortverzeichnis

