

PersonData Struktur: XML Spezifikation Dokumentation 14. Oktober 2004	Spezifikation
	pd20_doku
	Entwurf öffentlich

Bezeichnung	PersonData Struktur: XML Spezifikation– Dokumentation
Kurzbezeichnung	pd20_doku
Version	Version 2.0
Dokumentklasse	Spezifikation
Dokumentstadium	Entwurf öffentlich
Kurzbeschreibung	Diese Dokumentation beschreibt die PersonData XML Struktur version 2.0 für Domain Experten und Techniker.
Zielgruppe	XML Entwickler Systemanalysten Business Domain Experten
Autor	Larissa Naber larissa.naber@cio.gv.at
Editor	Arno Hollosi arno.hollosi@cio.gv.at
Arbeitsgruppe	Kommunikations Architektur Stabsstelle IKT Strategie Bundes
	

Inhaltsverzeichnis

1	Einleitung	1
1.1	Änderungen gegenüber Version 1.0	1
1.2	Spezifikationsaufbau	1
1.2.1	Kürzest Einführung in XML – Wie interpretiert man die Graphiken richtig?	2
2	PersonData 2.0 für Domain Experten	5
2.1	Root Elemente	5
2.1.1	Personendaten	5
2.1.2	Natürliche Personen	8
2.1.3	Personennamen	13
2.1.4	Nicht-natürliche Personen	16
2.1.5	Bankverbindungen	19
2.1.6	Adressen	22
2.1.7	Postadressen	23
2.1.8	Typisierte Postadressen	28
2.1.9	Internetadressen	29
2.1.10	Telefonadressen	31
2.1.11	Identifikationsdaten	35
2.1.12	Einfache Identifikationsdaten	37
2.2	Alternative Root Elemente	39
2.2.1	Kompakte Personendaten	40
2.2.2	Kompakte natürliche Person	43
2.2.3	Kompakte nicht natürliche Person	46
2.2.4	Kompakter Personenname	49
2.2.5	Kompakte Postadresse	52
3	PersonData 2.0 für Entwickler/Programmierer	55
3.1	Voraussetzungen	55
3.2	Technische Änderungen gegenüber Version 1.0	55
3.3	“Blaupausenkonzept” – myPersonData	56
3.3.1	Vorgehensweise	56
3.3.2	Beispiel 1	57
3.3.3	Beispiel 2	59
3.4	CompactPersonData Ansatz	69

3.5	“Atomare Elemente” Ansatz	69
3.6	Alternative Telekom Elemente	69
3.7	Welcher Ansatz für welches Problem?	72
3.7.1	Text- oder Repräsentationslastige Anwendungen	72
3.7.2	Webservices	72

Dieses Dokument verwendet die Schlüsselwörter MUSS, DARF NICHT, ERFORDERLICH, SOLLTE, SOLLTE NICHT, EMPFOHLEN, DARF, und OPTIONAL zur Kategorisierung der Anforderungen. Diese Schlüsselwörter sind analog zu ihren englischsprachigen Entsprechungen MUST, MUST NOT, REQUIRED, SHOULD, SHOULD NOT, RECOMMENDED, MAY, und OPTIONAL zu handhaben, deren Interpretation in RFC 2119 festgelegt ist.

Kapitel 1

Einleitung

Die `PersonData` Struktur soll es ermöglichen, alle Aspekte von Personendaten wie zum Beispiel Name, Geburtsdatum, Adresse oder Telefonnummern zu erfassen. Die `PersonData` Struktur kann mit natürlichen und nicht natürlichen Personen umgehen. Bei den Adressen stehen Postadressen, typisierte Postadressen, Internet Adressen und Telefonadressen zur Auswahl. 5

1.1 Änderungen gegenüber Version 1.0

Die Version 2.0 ist komplett rückwärtskompatibel zur Version 1.0. Version 2.0 bringt neue Datentypen, z.B. für ZMR konforme Adressen, typische österreichische Identifikationsnummern (Sozialversicherungsnummer, Matrikelnummer, ...) und kompakte Varianten der `PersonData` Struktur für den Einsatz in Webservices. 10

Weitere Änderungen betreffen technische Details des internen Datenmodells und sind im Kapitel für Entwickler zusammengefasst (siehe Abschnitt 3.2).

1.2 Spezifikationsaufbau 15

Die `PersonData` Struktur ist eine Komplexe XML Struktur und bringt außer den normativen Schemata noch eine Vielzahl von Sekundärinformation mit.

Für den ersten Überblick bietet sich die Überblicksgraphik `pd_map-pdf` an. Detaillierte Informationen finden sich in diesem Handbuch (`pd_doku.pdf`). Das Kapitel 2 bietet einen Überblick über alle wichtigen Elemente und ihre Abhängigkeiten und ist für Domainexperten und Entwickler gleichermaßen geeignet. Kapitel 3 beleuchtet die Einsatzmöglichkeiten der `PersonData` Struktur in gängigen E-Government Situationen. 20

Nur die Schemata selbst sind normativ. Bei Abweichung zwischen Handbuch und Schema zählt das Schema.

Dokument	Zielgruppe
Mindmap pd_map.pdf	Alle Bietet schnellen graphischen Überblick über alle Hauptelemente
Elementübersicht pd_allelements.xls	Alle Alternative Sicht auf alle Elemente. Elemente sind ihren Containern zugeordnet und auf mehrere Tabellenblätter verteilt
Äquivalenzliste pd_aequiv.pdf	Alle die mit zweisprachigen Dateien hantieren Bietet alphabetische Gegenüberstellung von deutschen und englischen Elementnamen als de ↔ en und en ↔ de Version
Handbuch pd_doku.pdf (dieses Dokument)	Alle Handbuch mit Hintergrundinformation und Abschnitten für Domainexperten und Entwickler
Schemata pd20_en.xsd pd20_de.xsd pd20_bi.xsd	Entwickler

Tabelle 1.1: Teildokumente und Zielgruppen

1.2.1 Kürzest Einführung in XML – Wie interpretiert man die Graphiken richtig?

25

Dieser Abschnitt stellt XML und XML Schemata kurz vor und soll Nicht-Technikern das notwendige Wissen zum Verstehen der Diagramme geben.

XML ist eine Technologie zum spezifizieren so genannter Markupssprachen. Markupssprachen dienen dazu die eigentliche Information gemeinsam mit Bezeichnern (Kategoriennamen) für diese Information gespeichert wird. Das sieht man am einfachsten an einem Beispiel.

30

```
<FirstName>Larissa</FirstName>
<LastName>Naber</LastName>
```

Beispiel 1: Eine einfache Struktur für Vor- und Nachnamen

Diese Markupssprache hat zwei sog. *Elemente* - `FirstName` und `FamilyName` - die jeweils einen Namensteil aufnehmen können. Welche Elemente an welcher Stelle möglich sind (also eine Art "Grammatik") wird in einem *Schema* (XSD) festgelegt. Die `PersonData`Struktur ist ein solches Schema.

35

Elemente können nicht nur Text sondern auch andere Elemente beinhalten

```
<PersonName>
  <FirstName>Larissa</FirstName>
  <LastName>Naber</LastName>
</PersonName>
```

Beispiel 2: Eine einfache Struktur für Personennamen

Das Element **PersonName** umschließt die bereits bekannten Elemente **FirstName** und **FamilyName**, die ihrerseits den eigentlichen Text beinhalten. In einer gebräuchlichen graphischen Umschreibung des Schemas stellt sich das obige Beispiel so dar:

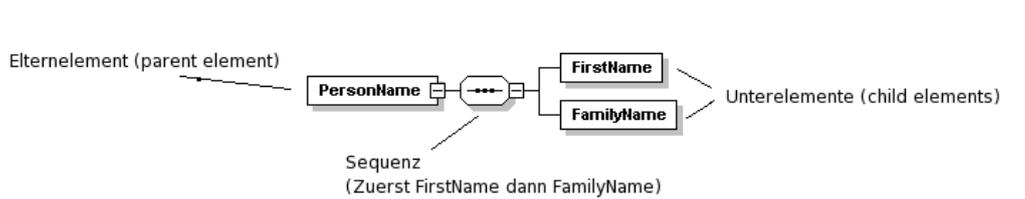


Abbildung 1.1: Sehr einfaches Schema für Personennamen

Elemente können genau einmal oder auch mehrfach vorkommen, oder optional sein. Viele Leute besitzen mehr als einen Vornamen und manche haben Titel oder andere Namenszusätze. Die rudimentäre Personennamenstruktur die wir in den letzten Beispielen wird nun erweitert: es soll möglich sein mehrere Vornamen und einen optionalen Namensaffix anzugeben. Graphisch würde das so aussehen:

40

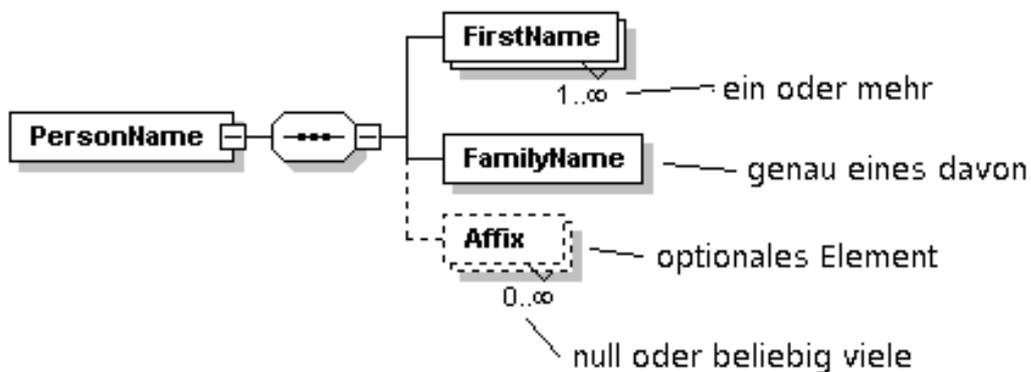


Abbildung 1.2: Ein etwas komplexeres Schema für Personennamen: optional und wiederholbare Elemente

Zugehörige Dokumente könnten so aussehen

45

```
<PersonName>
  <FirstName>Larissa</FirstName>
  <FirstName>Magaretha</FirstName>
  <LastName>Naber</LastName>
</PersonName>
```

Beispiel 3: Personenname mit zweitem Vornamen

```
<PersonName>
  <FirstName>Larissa</FirstName>
  <FirstName>Magaretha</FirstName>
  <LastName>Naber</LastName>
  <Affix>Dr.</Affix>
</PersonName>
```

Beispiel 4: Personenname mit zweitem Vornamen und Affix

Es ist auch möglich in einem Schema alternative Elemente zur Auswahl anzubieten. Angenommen wir wollen ein Schema für Kontaktdaten aufbauen. Wir brauchen dazu eine Adresse und einen Namen. Der Name kann allerdings auch ein Firmenname sein. Graphisch sieht das dann so aus:

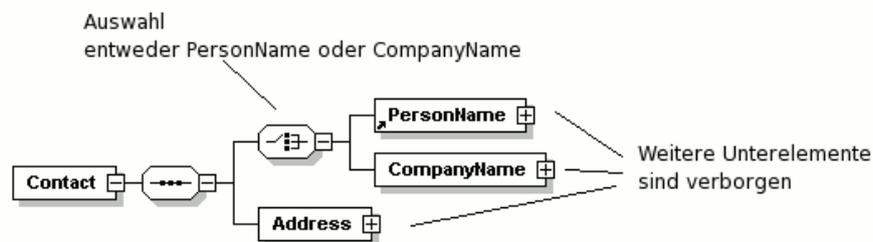


Abbildung 1.3: Ein Schema für Kontaktdaten: alternative Unterlemente

Das Element **Contact** hat ein Unterelement **Address** und ein weiteres Unterelement, das entweder ein **PersonName** oder ein **CompanyName** Element ist. Die Struktur des **Address** oder **CompanyName** Elements ist für dieses Beispiel nicht relevant.

Kapitel 2

PersonData 2.0 für Domain Experten

2.1 Root Elemente

55

2.1.1 Personendaten

Englische Version: PersonData

Deutsche Version: PersonenDaten

Beschreibung: Container für diverse Informationen über Personen

Kein Elternelement

60

Englisch	Deutsch	Erklärung
Identification	Identifikation	(siehe Abschnitt 2.1.11)
AbstractSimple- Identification	AbstractSimple- Identifikation	(siehe Abschnitt 2.1.12)
CompactPhysical- Person	Kompakte- NatuerlichePerson	(siehe Abschnitt 2.2.2)
CompactCorporate- Body	KompakteNicht- NatuerlichePerson	(siehe Abschnitt 2.2.3)
PhysicalPerson	NatuerlichePerson	(siehe Abschnitt 2.1.2)
CorporateBody	NichtNatuerliche- Person	(siehe Abschnitt 2.1.4)
Address	Address	(siehe Abschnitt 2.1.6)
InternetAddress	InternetAdresse	(siehe Abschnitt 2.1.9)
CompactPostal- Address	KompaktePost- Adresse	(siehe Abschnitt 2.2.5)
PostalAddress	PostAdress	(siehe Abschnitt 2.1.7)
TelephoneAddress	TelefonAdresse	(siehe Abschnitt 2.1.10)
TypedPostalAddress	TypisiertePost- Adresse	(siehe Abschnitt 2.1.8)
dsig:Signature	dsig:Signature	Internationaler Standard für digitale Signaturen, nur von technischem Interesse
AdditionalData	Zusatzdaten	Container für Zusatzdaten zur Person

Tabelle 2.1: Unterelemente von PersonData und PersonenDaten Elementen

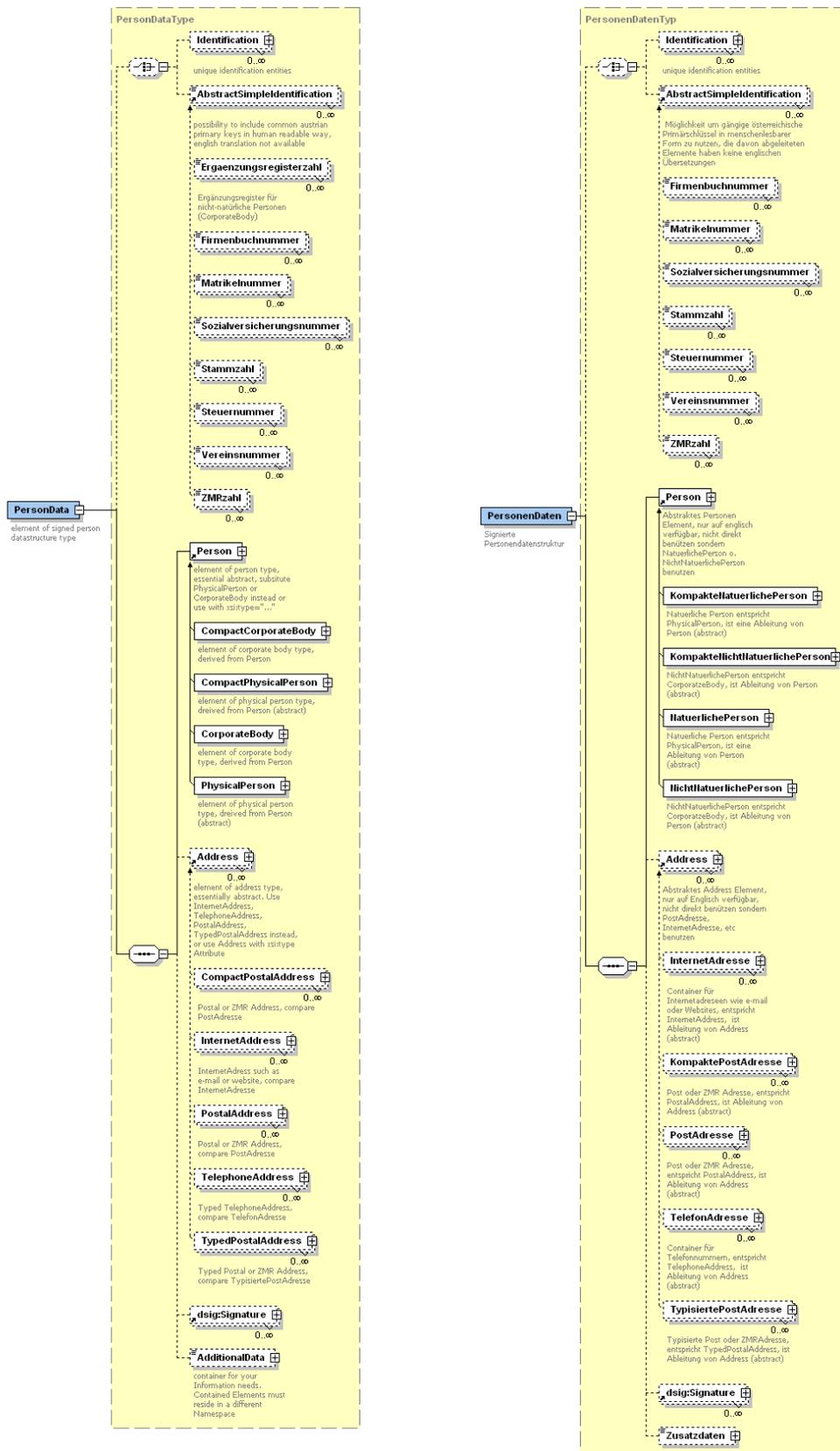


Abbildung 2.1: Unterlemente des PersonData und PersonenDaten Elements

2.1.2 Natürliche Personen

Englische Version: PhysicalPerson

Deutsche Version: NatuerlichePerson

Beschreibung: Container für diverse Informationen über Natürliche Personen

Parent Elemente (en, de): PersonData, PersonenDaten (siehe Abschnitt 2.1.1)

65

Staatsangehoerigkeit

Englische Version: Nationality

Deutsche Version: Staatsangehoerigkeit

Beschreibung: Container für die genaue Beschreibung der Staatsangehörigkeit

Parent Elemente (en, de): PhysicalPerson, NatuerlichePerson (siehe Abschnitt 2.1.2)

70

Verwandter

Englische Version: RelatedPerson

Deutsche Version: Verwandter

Beschreibung: Container für die genaue Beschreibung des Verwandtschaftsgrades und der Person

Parent Elemente (en, de): PhysicalPerson, NatuerlichePerson (siehe Abschnitt 2.1.2)

75

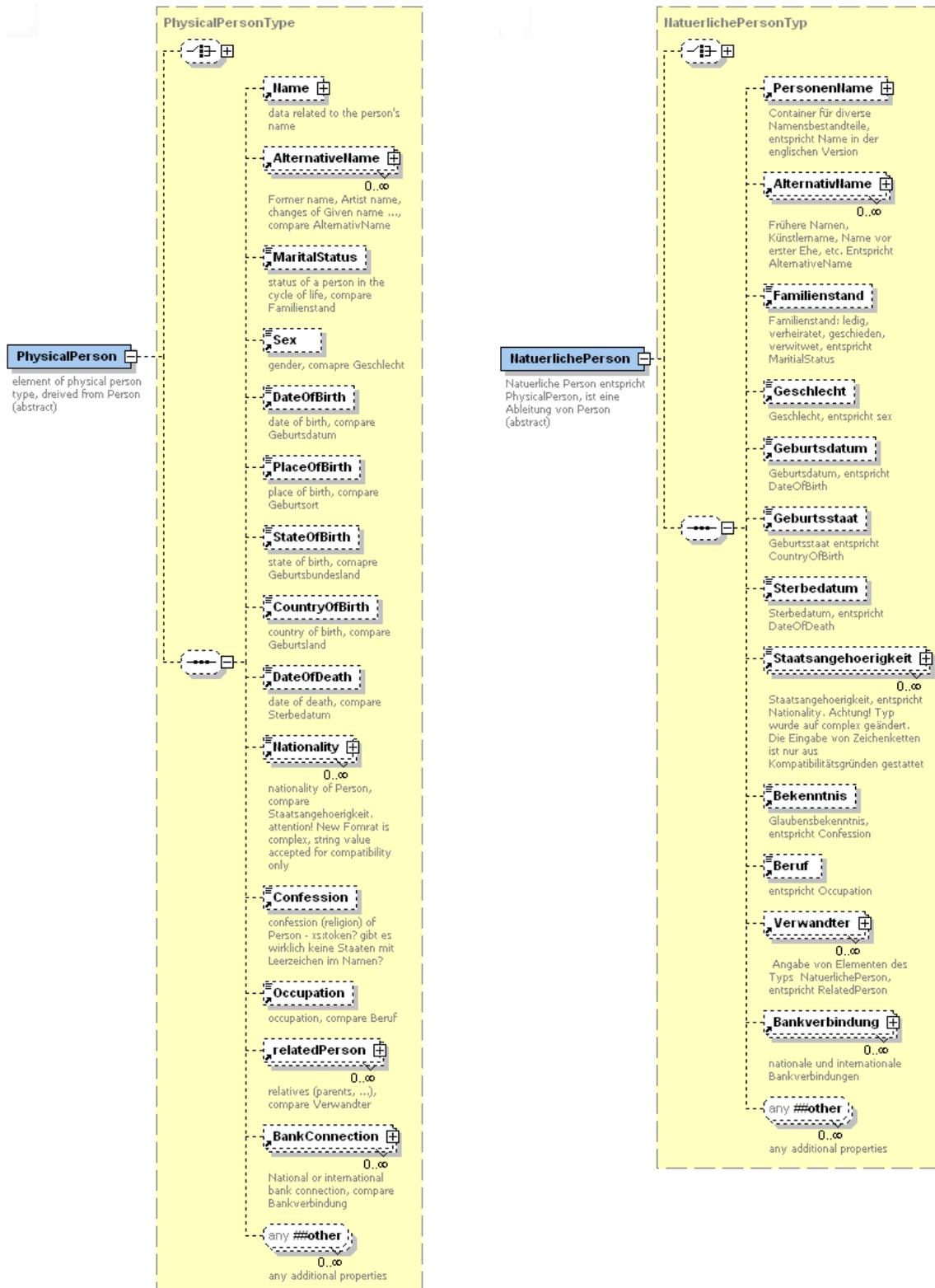


Abbildung 2.2: Unterlemente des PhysicalPerson und NatuerlichePerson Elements

Englisch	Deutsch	Erklärung
Identification	Identifikation	Container für diverse Identifikationsmerkmale. (siehe Abschnitt 2.1.11)
AbstractSimple- Identification	AbstractSimple- Identification	Sammelbegriff für verschiedene in Österreich häufig vorkommende Identifikationsnummern. (siehe Abschnitt 2.1.12)
Name	PersonenName	Container für diverse Namensbestandteile. (siehe Abschnitt 2.1.3)
AlternativeName	AlternativName	Frühere Namen, Künstlername, Name vor ersten Ehe, vom selben Typ wie Personennamen. (siehe Abschnitt 2.1.3) Attribute: type/typ
MartialStatus	Familienstand	Mögliche Werte: (single/ledig, married/verheiratet, divorced/geschieden, widowed/verwitwet)
Sex	Geschlecht	Mögliche Werte: (male/männlich, female/weiblich, unknown/unbekannt)
DateOfBirth	Geburtsdatum	Geburtsdatum: Tag und/oder Monat können ausgelassen werden.
PlaceOfBirth	Geburtsort	Einfache Angabe des Geburtsortes.
StateOfBirth	Geburtsbundesland	Geburtsbundesland
CountryOfBirth	Geburtsstaat	Geburtsstaat
DateOfDeath	Sterbedatum	Sterbedatum: Tag und/oder Monat können ausgelassen werden.
Nationality	Staatsangehörigkeit	(siehe Unterpunkt 2.1.2)
Confession	Bekenntnis	Religiöses Bekenntnis
Occupation	Beruf	Name des Berufs, der ausgeübt wird.
RelatedPerson	Verwandter	Angabe von Verwandten (siehe Unterpunkt 2.1.2)
BankConnection	Bankverbindung	Angaben über die Bankverbindung (siehe Abschnitt 2.1.5)
##other	##other	Möglichkeit weitere Eigenschaften einzufügen, die bisher nicht angegeben wurden.

Tabelle 2.2: Unterelemente von PhysicalPerson und NatuerlichePerson Elementen

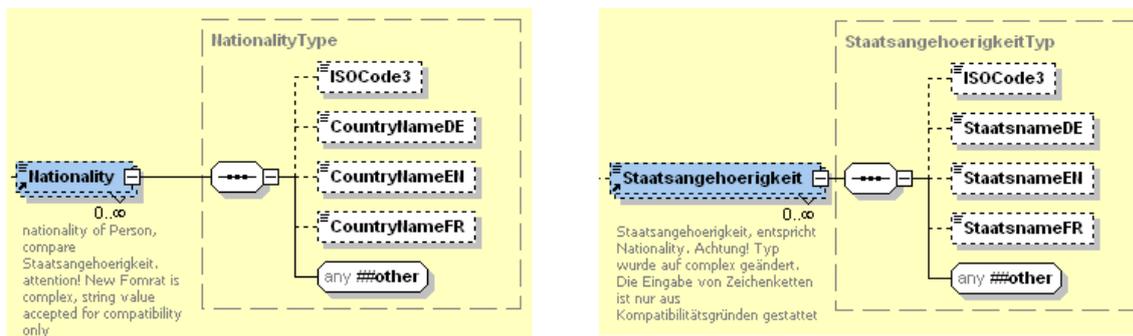


Abbildung 2.3: Unterlemente des Nationality und Staatsangehoerigkeits Containers

Englisch	Deutsch	Erklärung
ISOCode3	ISOCode3	Staatencode aus 3 Buchstaben
CountryNameDE	StaatsnameDE	Texteingabe des Landesnamens in Deutsch
CountryNameEN	StaatsnameEN	Texteingabe des Landesnamens in Englisch
CountryNameFR	StaatsnameFR	Texteingabe des Landesnamens in Französisch
##other	##other	Möglichkeit weitere Eigenschaften einzufügen, die bisher nicht angegeben wurden.

Tabelle 2.3: Unterelemente von Nationality und Staatsangehoerigkeit Elementen

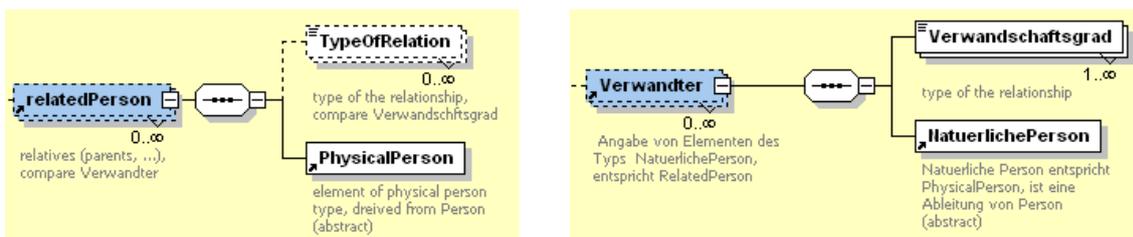


Abbildung 2.4: Unterlemente des RelatedPerson und Verwandter Containers

Englisch	Deutsch	Erklärung
TypeOfRelation	Verwandtschaftsgrad	Verwandtschaftsgrad, der auch aus einer Liste ausgewählt werden kann.
PhysicalPerson	NatuerlichePerson	Entspricht dem Typ NatuerlichePerson und ist eine Ableitung dieser.

Tabelle 2.4: Unterelemente von RelatedPerson und Verwandter Elementen

Beispiele

```
<NaturlichePerson>
  <PersonenName>
    <Vorname>Max</Vorname>
    <Familiename>Mustermann</Familiename>
  </PersonenName>
  <Geschlecht>männlich</Geschlecht>
  <Beruf>Drucker</Beruf>
</NaturlichePerson>
```

Beispiel 5: Herr Max Mustermann ist Drucker

```
<NaturlichePerson>
  <PersonenName>
    <Vorname>Emilie</Vorname>
    <Familiename>Mustermann</Familiename>
  </PersonenName>
  <AlternativName Type="NameVorErsterEhe">
    <Familiename>Huber</Familiename>
  </AlternativName>
  <Familienstand>verheiratet</Familienstand>
  <Geschlecht>weiblich</Geschlecht>
  <Geburtsdatum>13.2.1963</Geburtsdatum>
  <Geburtsort>Wien</Geburtsort>
  <Geburtsstaat>Österreich</Geburtsstaat>
  <Bekenntnis>röm.-kat.</Bekenntnis>
  <Beruf>Hausfrau</Beruf>
</NaturlichePerson>
```

Beispiel 6: Emilie Mustermann mit diversen Zusatzinformationen

2.1.3 Personennamen

Englische Version: Name

Deutsche Version: PersonenName

80

Beschreibung: Container für diverse Bestandteile eines Personennamens

Parent Elemente (en, de): PhysicalPerson, NatuerlichePerson (siehe Abschnitt 2.1.2)

Bedingt durch die Geschichte der PersonDataStruktur, enthält dieser Container sehr viele Amerikanismen, die in Österreich/Europa eher ungebräuchlich sind und gemieden werden sollten.

85

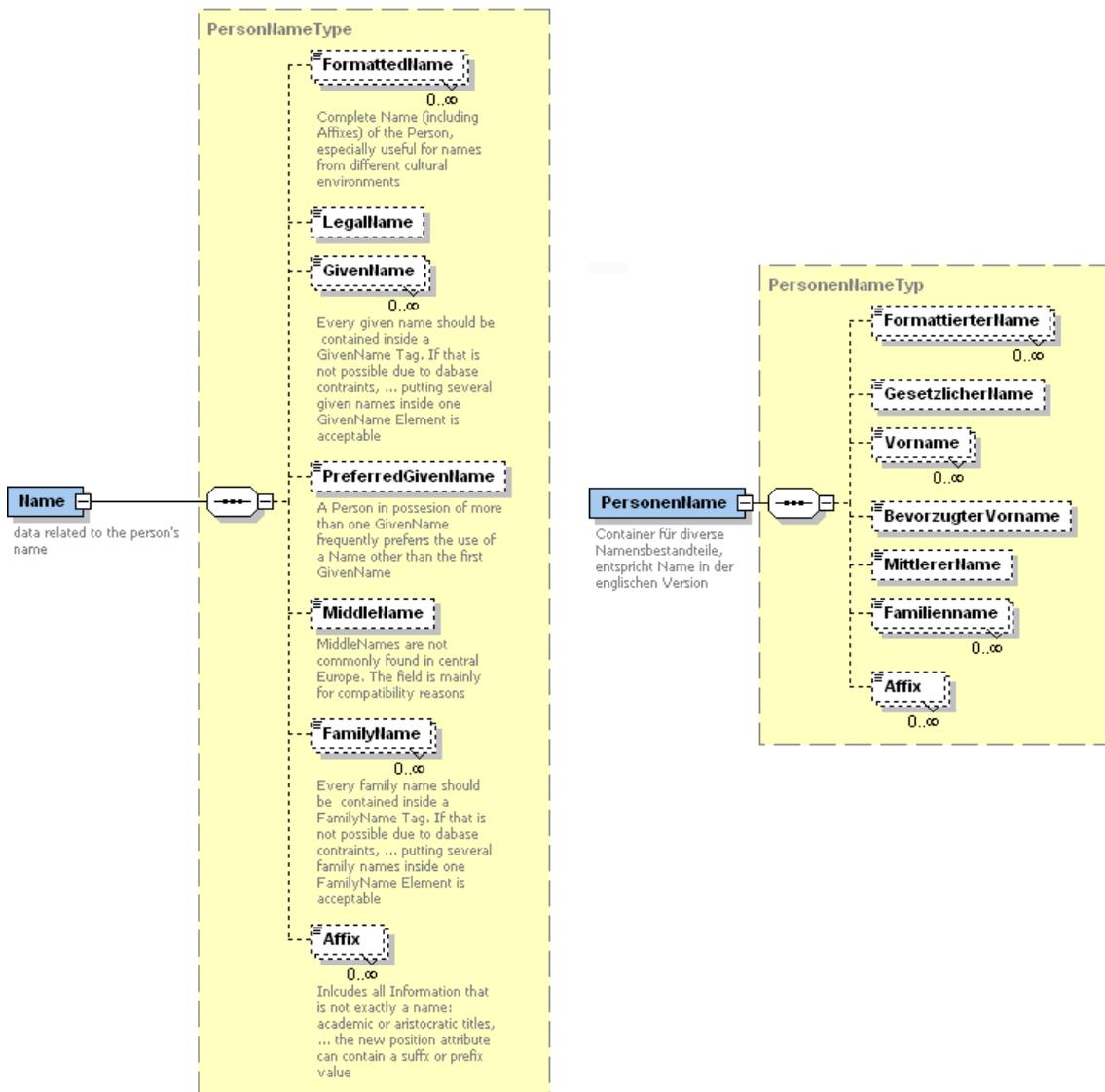


Abbildung 2.5: Unterlemente des Name und PersonenName Elements

Englisch	Deutsch	Erklärung
FormattedName	FormattierterName	Formatierter Name inklusive Prefix, Postfix. Besonders geeignet für Name aus anderen Sprachräumen. Attribute: type (presentation, legal/rechtlich, sortOrder/sortiert)
LegalName	GesetzlicherName	Amerikanismus, sollte nicht verwendet werden
GivenName	Vorname	Vorname, Jeden Vornamen in eigenes GivenName Tag einschließen
PreferedGivenName	BevorzugterVorname	Name der von der Person bevorzugt wird (nur bei mehreren Vornamen!)
MiddleName	MittlerName	Amerikanismus, sollte nicht verwendet werden
FamilyName	Familiename	Familiename, jeden Familiennamen der durch ein Leerzeichen getrennt ist, in einen eigenes FamilyName Tag einschließen. Attribute: primary/primaer (true/wahr, false/falsch, undefined/undefiniert) prefix
Affix	Affix	Pre- oder Postfix: z.B: akademische Titel, ... Attribute: type/typ (academicGrade/akademischerGrad, aristocraticPrefix/–, aristocraticTitle/Adelstitel, familyNamePrefix/FamiliennamenAffix, familyNameSuffix/–, –/Anrede, formOfAddress/– generation/Generation, qualification/Qualifikation) position (prefix, suffix)

Tabelle 2.5: Unterelemente von Name und PersonenName Elementen

Beispiele

```
<Name>
  <FirstName>Hans-Peter</FirstName>
  <PreferredGivenName>Laurenz</PreferredGivenName>
  <FamilyName>Maier</FamilyName>
</Name>
```

Beispiel 7: Hans-Peter Laurenz Maier, wird bevorzugt Laurenz genannt

```
<Name>
  <FirstName>Marie</FirstName>
  <FirstName>Sophie</FirstName>
  <FamilyName>Helsing</FamilyName>
  <Affix type="familyNamePrefix" position="prefix">van</Affix>
</Name>
```

Beispiel 8: Marie Sophie van Helsing

```
<Name>
  <FirstName>Alexander</FirstName>
  <FamilyName>Leiningen-Westerburg</FamilyName>
  <Affix type="academicGrade" position="prefix">Mag.</Affix>
  <Affix type="academicGrade" position="suffix">MAS</Affix>
</Name>
```

Beispiel 9: Mag. Alexander Leiningen-Westerburg, MAS

2.1.4 Nicht-natürliche Personen

Englische Version: CorporateBody

Deutsche Version: NichtNaturerlichePerson

Beschreibung: Container für diverse Informationen über Nicht-Natürliche Personen

Parent Elemente (en, de): PersonData, PersonenDaten (siehe Abschnitt 2.1.1)

90

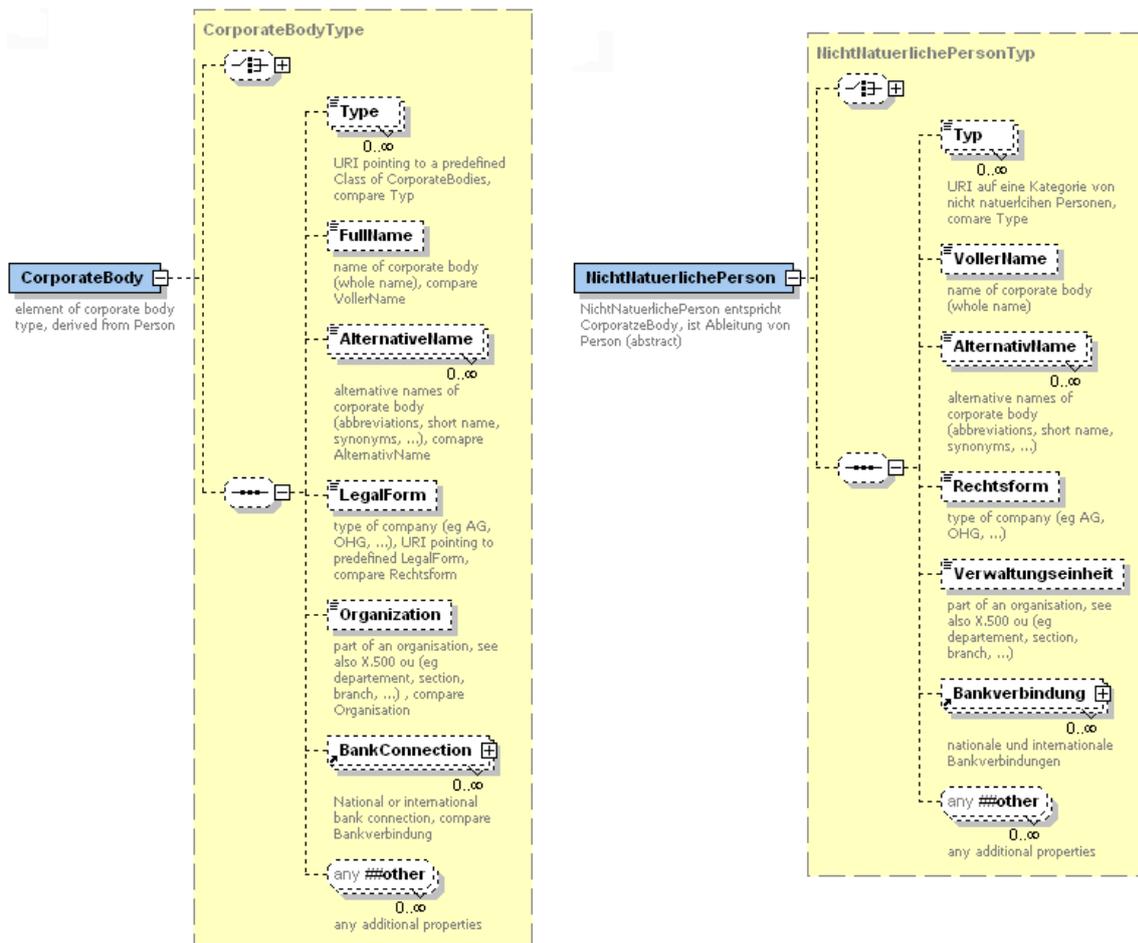


Abbildung 2.6: Unterlemente des CorporateBody und NichtNaturerlichePerson Elements

Englisch	Deutsch	Erklärung
Identification	Identifikation	Container für diverse Identifikationsmerkmale. (siehe Abschnitt 2.1.11)
AbstractSimple- Identification	AbstractSimple- Identification	Sammelbegriff für verschiedene in Österreich häufig vorkommende Identifikationsnummern. (siehe Abschnitt 2.1.12)
Type	Typ	URI auf eine Kategorie von nicht natuerlichen Personen.
Fullname	VollerName	Ganzer Name der nicht natuerlichen Person.
AlternativeName	AlternativName	Alternativen zum vollen Namen wie Synonyme, Abkürzungen,...
LegalForm	Rechtsform	Typ der nicht natuerlichen Person wie AG, OHG,...
Organization	Verwaltungseinheit	Teil der Organisation wie Sektion, Abteilung,...
BankConnection	Bankverbindung	Angaben über die Bankverbindung (siehe Abschnitt 2.1.5)
##other	##other	Möglichkeit weitere Eigenschaften einzufügen, die bisher nicht angegeben wurden.

Tabelle 2.6: Unterelemente von CorporateBody und NichtNatuerlichePerson Elementen

Beispiele

```
<NichtNatuerlichePerson>
  <VollerName>Druckerei Ganster</VollerName>
  <AlternativName>DruckGans</AlternativName>
  <AlternativName>Ganster Druck</AlternativName>
  <Rechtsform>AG</Rechtsform>
</NichtNatuerlichePerson>
```

Beispiel 10: Die Firma Druckerei Ganster wird oft auch DruckGans oder Ganster Druck genannt und ist eine AG.

```
<NichtNatuerlichePerson>
  <VollerName>Druckerei Ganster</VollerName>
  <AlternativName>DruckGans</AlternativName>
  <AlternativName>Ganster Druck</AlternativName>
  <Rechtsform>AG</Rechtsform>
  <Bankverbindung>
    <Inhaber>Druckerei Ganster AG</Inhaber>
    <BankName>Die Erste</BankName>
    <NationaleBankverbindung>
      <Kontonummer>01234567890</Kontonummer>
      <BLZ>12345</BLZ>
    </NationaleBankverbindung>
  </Bankverbindung>
</NichtNatuerlichePerson>
```

Beispiel 11: Die Firma Druckerei Ganster w.o. inklusive nationaler Bankverbindung

2.1.5 Bankverbindungen

Englische Version: BankConnection

Deutsche Version: Bankverbindung

Beschreibung: Container für Kontoinformationen

Parent Elemente (en, de): CorporateBody, NichtNatuerlichePerson (siehe Abschnitt 2.1.4)

Parent Elemente (en, de): PhysicalPerson, NatuerlichePerson (siehe Abschnitt 2.1.2)

95

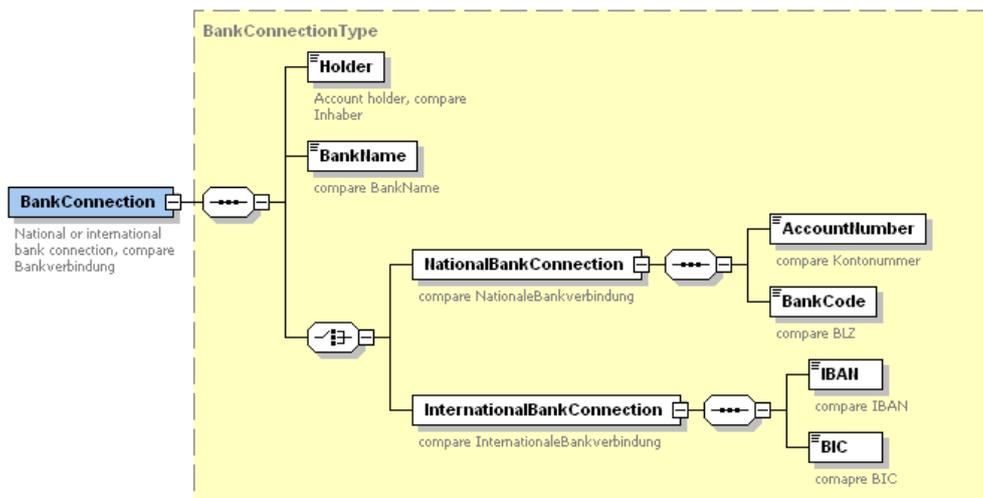


Abbildung 2.7: Unterlemente des BankConnection Containers

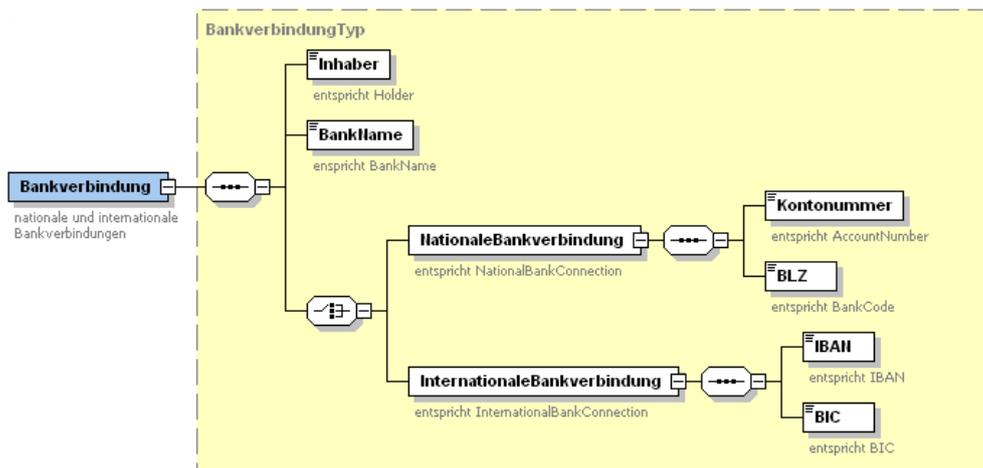


Abbildung 2.8: Unterlemente des Bankverbindung Containers

NationaleBankverbindung

Englische Version: NationalBankConnection

Deutsche Version: NationaleBankverbindung

100

Englisch	Deutsch	Erklärung
Holder	Inhaber	Name des Inhabers der Bankverbindung.
BankName	BankName	Name des Bankinstituts.
NationalBankConnection	NationaleBankverbindung	Typ für nationale Bankverbindungen.
InternationalBankConnection	InternationaleBankverbindung	Typ für internationale Bankverbindungen.

Tabelle 2.7: Unterelemente von BankConnection und Bankverbindung Elementen

Beschreibung: Container für genaue Banverbindungsangaben einer nationale Bank

Parent Elemente (en, de): BankConnection, Bankverbindung (siehe Abschnitt 2.1.5)

Englisch	Deutsch	Erklärung
AccountNumber	Kontonummer	Kontonummer als Zahl.
BankCode	BLZ	BLZ als Zahl.

Tabelle 2.8: Unterelemente des NationalBankConnection und NationaleBankverbindung Containers

InternationaleBankverbindung

105

Englische Version: InternationalBankConnection

Deutsche Version: InternationaleBankverbindung

Beschreibung: Container für genaue Banverbindungsangaben einer internationalen Bank

Parent Elemente (en, de): BankConnection, Bankverbindung (siehe Abschnitt 2.1.5)

Englisch	Deutsch	Erklärung
IBAN	IBAN	International Bank Account Number.
BIC	BIC	Bank Identifier Code.

Tabelle 2.9: Unterelemente des InternationalBankConnection und InternationaleBankverbindung Containers

Beispiele

110

```
<Bankverbindung>
  <Inhaber>Druckerei Ganster AG</Inhaber>
  <BankName>Die Erste</BankName>
  <NationaleBankverbindung>
    <Kontonummer>01234567890</Kontonummer>
    <BLZ>12345</BLZ>
  </NationaleBankverbindung>
</Bankverbindung>
```

Beispiel 12: Nationale Bankverbindung - Konto der Firma Druckerei Ganster bei der Ersten

```
<Bankverbindung>
  <Inhaber>Druckerei Ganster AG</Inhaber>
  <BankName>Commerzbank</BankName>
  <InternationaleBankverbindung>
    <IBAN>DE53 3004 0000 0123 4567 00</IBAN>
    <BIC>COBADEFF</BIC>
  </InternationaleBankverbindung>
</Bankverbindung>
```

Beispiel 13: Internationale Bankverbindung - Konto der Firma Druckerei Ganster bei der Commerzbank in Deutschland

2.1.6 Adressen

Die Adresse wird über verschiedene Untertypen definiert. Für die Eingabe von Adressen sollten diese Untertypen gewählt werden und nicht direkt das Adressen-Element. So gibt es die

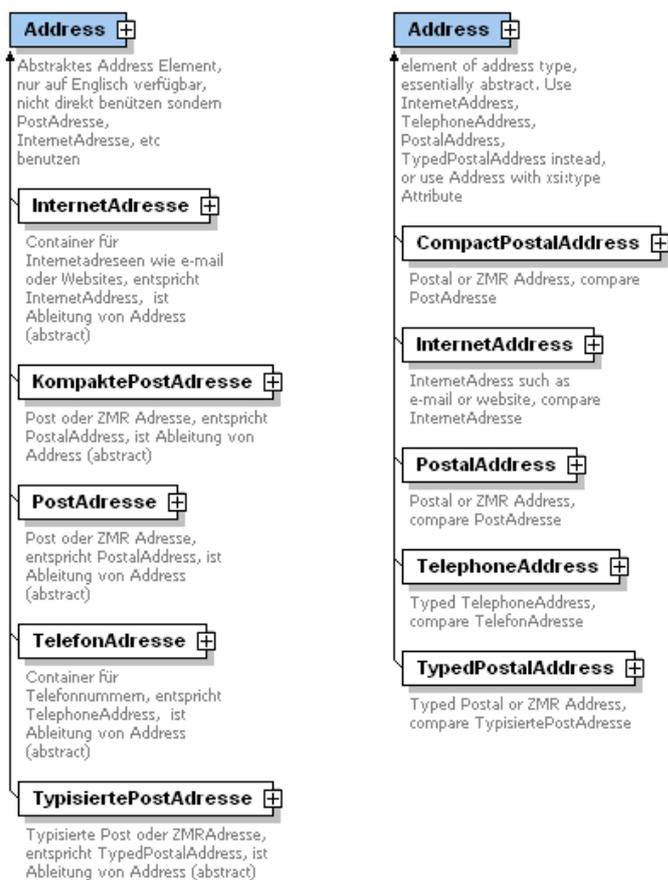


Abbildung 2.9: Ableitungen des Address Elements

Englisch	Deutsch	Erklärung
InternetAddress	InternetAdresse	(siehe Abschnitt 2.1.9)
CompactPostalAddress	KompaktePostadresse	(siehe Abschnitt 2.2.5)
PostalAddress	PostAdresse	(siehe Abschnitt 2.1.7)
TelephoneAddress	TelefonAdresse	(siehe Abschnitt 2.1.10)
TypedPostalAddress	TypisiertePostAdresse	(siehe Abschnitt 2.1.8)

Tabelle 2.10: Ableitungen des Address Elements

2.1.7 Postadressen

115

Englische Version: PostalAddress

Deutsche Version: PostAdresse

Beschreibung: Container für diverse Adressbestandteile

Attribute: type

Parent Elemente (en, de): PersonData, PersonenDaten (siehe Abschnitt 2.1.1)

120

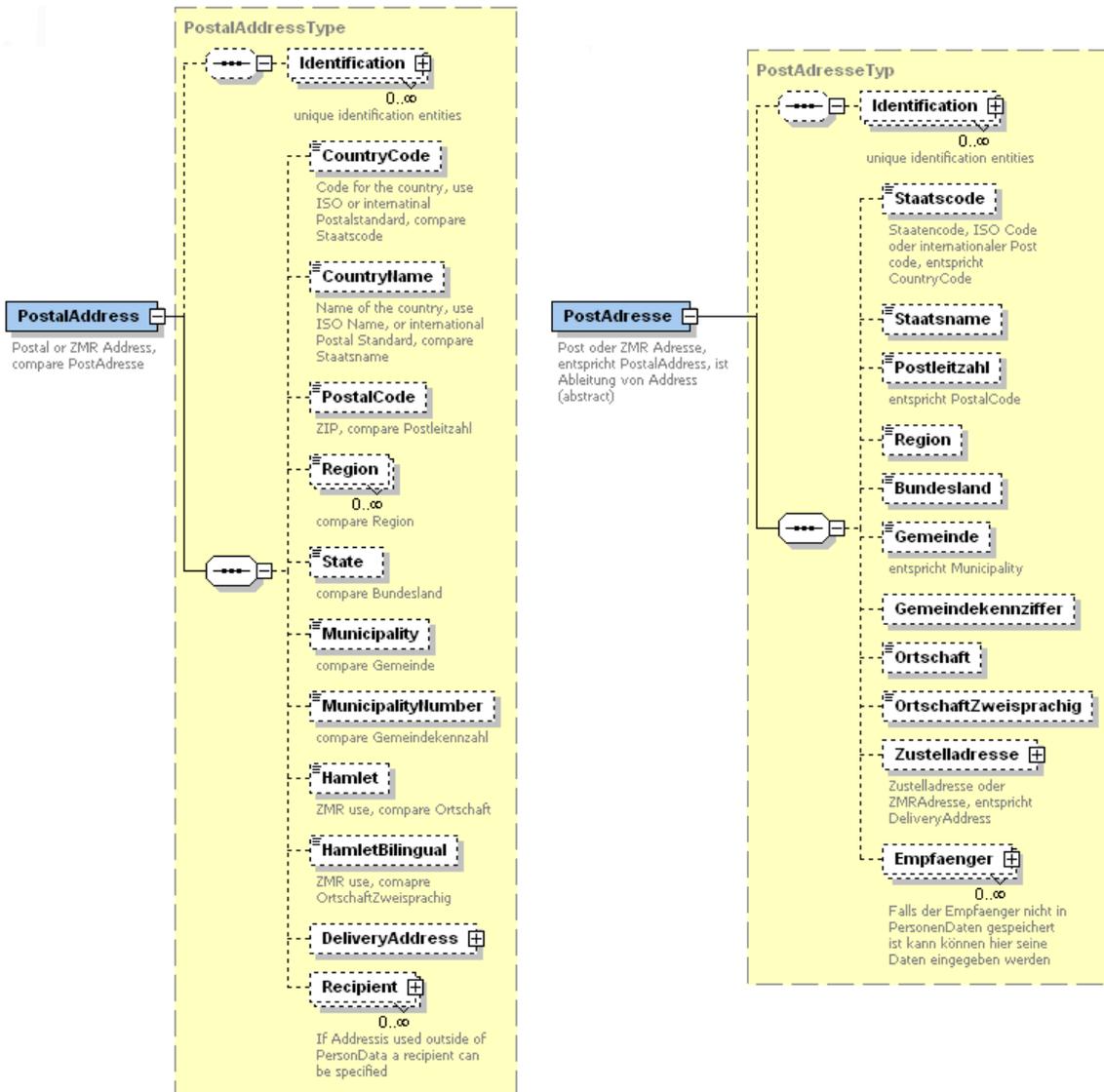


Abbildung 2.10: Unterlemente des PostalAddress und PostAdresse Containers

Zustelladressen

Englische Version: DeliveryAddress

Deutsche Version: Zustelladresse

Englisch	Deutsch	Erklärung
Identification	Identifikation	Container für diverse Identifikationsmerkmale. (siehe Abschnitt 2.1.11)
CountryCode	Staatscode	Staatencode, ISO Code oder internationaler Postcode.
CountryName	Staatsname	Staatsname, ISO Namen benutzen.
PostalCode	Postleitzahl	Postleitzahl
Region	Region	Region
State	Bundesland	Bundesland (Wien, Niederösterreich, Burgenland, Oberösterreich, Steiermark, Salzburg, Kärnten, Tirol, Vorarlberg)
Municipality	Gemeinde	Gemeinde
Municipality-Number	Gemeindekennziffer	Gemeindekennziffer
Hamlet	Ortschaft	Ortschaft
HamletBilingual	Ortschaft-Zweisprachig	OrtschaftZweisprachig
DeliveryAddress	Zustelladresse	Zustelladresse (siehe Unterpunkt 2.1.7)
Recipient	Empfaenger	Empfaenger (siehe Unterpunkt 2.1.7)

Tabelle 2.11: Unterelemente von PostalAddress und PostAdresse Elementen

Beschreibung: Container für genaue Strassenadressangaben

Parent Elemente (en, de): PostAddress, PostAdresse (siehe Abschnitt 2.1.7)

125

AdressRegisterEintrag

Englische Version: AddressRegisterEntry

Deutsche Version: AdressRegisterEintrag

Beschreibung: Container für die Identifizierung der Adresse

Parent Elemente (en, de): PostAddress, PostAdresse (siehe Abschnitt 2.1.7)

130

Empfänger

Englische Version: Recipient

Deutsche Version: Empfaenger

Beschreibung: Container für zusätzliche Empfängerangaben innerhalb einer Postanschrift.

Parent Elemente (en, de): PostAddress, PostAdresse (siehe Abschnitt 2.1.7)

135

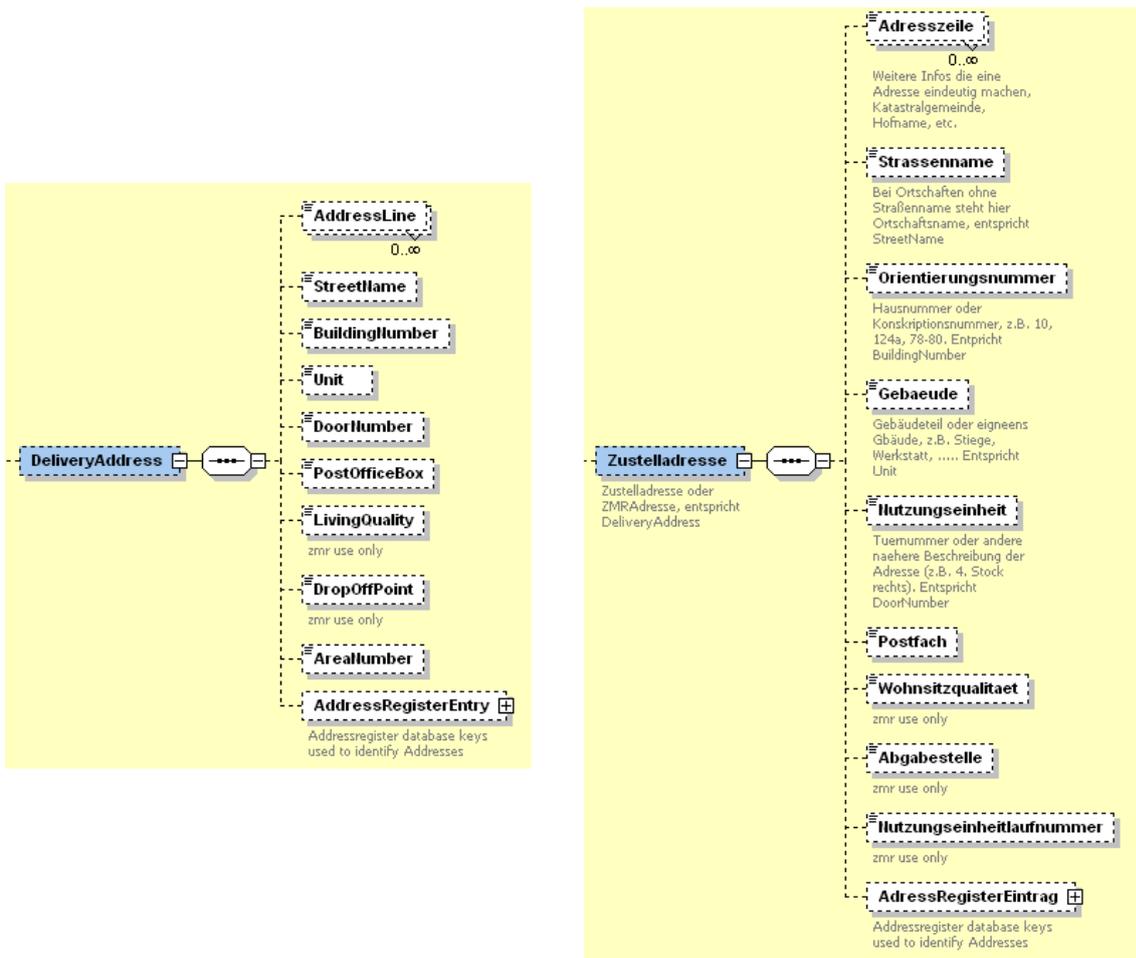


Abbildung 2.11: Unterlemente des DeliveryAddress und Zustelladresse Containers



Abbildung 2.12: Unterlemente des AddressRegisterEntry und AdressRegisterEintrag Containers



Abbildung 2.13: Unterlemente des Recipient und des Empfaenger Containers

Englisch	Deutsch	Erklärung
AddressLine	Adresszeile	Weiter Infos, die eine Adresse eindeutig machen, Katastralgemeinde, Hofname, ...
StreetName	Strassenname	Bei Ortschaften ohne Straßennamen steht hier Ortschaftsname
BuildingNumber	Orientierungsnummer	Hausnummer oder Konskriptionsnummer, z.B. 10, 124a, 78–80
Unit	Gebaeude	Gebäudeteil oder eigenes Gebäude, z.B. Stiege, Werkstatt,...
DoorNumber	Nutzungseinheit	Tuernummer oder andere naehere Beschreibung der Adresse, z.B. 4. Stock rechts
PostOfficeBox	Postfach	Postfach
LivingQuality	Wohnsitzqualitaet	Wohnsitzqualität
DropOffPoint	Abgabestelle	Abgabestelle
AreaNumber	Nutzungseinheit- laufnummer	Nutzungseinheitlaufnummer
AddressRegister- Entry	AdressRegister- Eintrag	AdressRegisterEintrag (siehe Unterpunkt 2.1.7)

Tabelle 2.12: Unterelemente von DeliveryAddress und Zustelladresse Elementen

Englisch	Deutsch	Erklärung
AddressCode	Adresscode	Adresscode
SubCode	Subcode	Subcode
ObjectNumber	Objektnummer	Objektnummer

Tabelle 2.13: Unterelemente von AddressRegisterEntry und AdressRegisterEintrag Elementen

Englisch	Deutsch	Erklärung
PersonName	PersonenName	Personen Name (siehe Abschnitt 2.1.3)
AdditionalText	Zusatztext	Zusatzinformationen
Organization	Verwaltungseinheit	Organisationseinheit
OrganizationName	Unternehmen	Name der Organisation

Tabelle 2.14: Unterelemente von Recipient und Empfaenger Elementen

Beispiele

```
<PostAdresse type="Anschrift">
  <Postleitzahl>1010</Postleitzahl>
  <Bundesland>Wien</Bundesland>
  <Gemeinde>Wien</Gemeinde>
  <Zustelladresse>
    <Strassenname>Ringstraße</Strassenname>
    <Orientierungsnummer>1</Orientierungsnummer>
    <Nutzungseinheit>4. Stock rechts</Nutzungseinheit>
  </Zustelladresse>
</PostAdresse>
```

Beispiel 14: Eine mögliche Postanschrift in 1010 Wien, Ringstrasse 1 mit näherer Erklärung

```
<PostAdresse type="Anschrift">
  <Postleitzahl>1010</Postleitzahl>
  <Bundesland>Wien</Bundesland>
  <Gemeinde>Wien</Gemeinde>
  <Zustelladresse>
    <Strassenname>Ringstraße</Strassenname>
    <Orientierungsnummer>1</Orientierungsnummer>
    <Nutzungseinheit>4. Stock rechts</Nutzungseinheit>
  </Zustelladresse>
  <Empfaenger>
    <PersonenName>
      <Vorname>Max</Vorname>
      <Familiename>Mustermann</Familiename>
    </PersonenName>
    <Zusatztext>Firmengründer</Zusatztext>
    <Verwaltungseinheit>Personalbüro</Verwaltungseinheit>
    <Unternehmen>Druckerei Ganster</Unternehmen>
  </Empfaenger>
</PostAdresse>
```

Beispiel 15: Postanschrift 1010 Wien, Ringstrasse 1 mit näherer Erklärung und Angaben zum Empfänger

2.1.8 Typisierte Postadressen

Englische Version: TypedPostalAddress

Deutsche Version: TypisiertePostAdresse

Beschreibung: Container für diverse Adressbestandteile

140

Parent Elemente (en, de): PersonData, PersonenDaten (siehe Abschnitt 2.1.1)

Dieser Typ unterscheidet sich nur unwesentlich vom gewöhnlichen Postadresstyp: der Unterschied besteht in einem Element **Type** oder **Typ** das die Art der Adresse näher beschreibt und als erstes Element im Container eingefügt werden muß

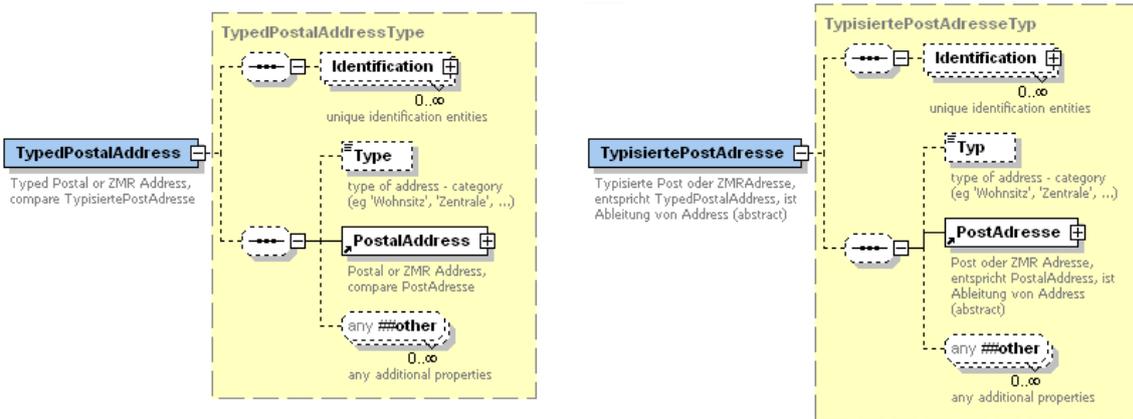


Abbildung 2.14: Unterlemente des TypedPostalAddress und TypisiertePostAdresse Containers

Englisch	Deutsch	Erklärung
Identification	Identifikation	Container für diverse Identifikationsmerkmale. (siehe Abschnitt 2.1.11)
Type	Typ	Nähere Beschreibung der Adresse
PostalAddress	PostAdresse	Postadresse Attribute: type (postOfficeBoxAddress/Postfachadresse, streetAddress/Anschrift, militaryAddress/MilitaerischeAdresse, undefined/undefiniert) (siehe Abschnitt 2.1.7)
##other	##other	Möglichkeit weitere Eigenschaften einzufügen, die bisher nicht angegeben wurden.

Tabelle 2.15: Unterlemente von Recipient und Empfaenger Elementen

2.1.9 Internetadressen

145

Englische Version: InternetAddress

Deutsche Version: InternetAdresse

Beschreibung: Container für diverse Internetadressbestandteile wie E-Mail oder Webadressen

Parent Elemente (en, de): PersonData, PersonenDaten (siehe Abschnitt 2.1.1)

150

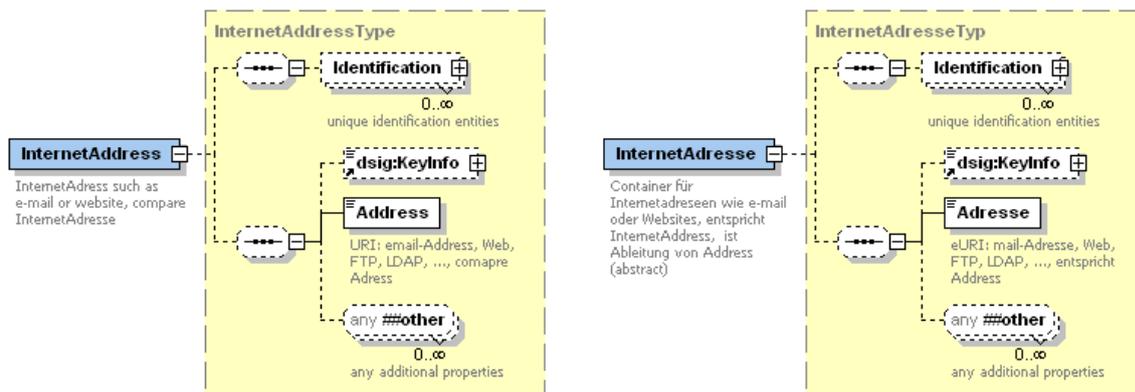


Abbildung 2.15: Unterlemente des InternetAddress und InternetAdresse Containers

Englisch	Deutsch	Erklärung
Identification	Identifikation	Container für diverse Identifikationsmerkmale. (siehe Abschnitt 2.1.11)
dsig:KeyInfo	dsig:KeyInfo	Internationaler Standard für digitale Signaturen, nur von technischem Interesse
Address	Adresse	eURI: mailAdresse, Web, FTP, LDAP,...
##other	##other	Möglichkeit weitere Eigenschaften einzufügen, die bisher nicht angegeben wurden.

Tabelle 2.16: Unterelemente von InternetAddress und InternetAdresse Elementen

Beispiele

```
<InternetAdresse>  
  <Adresse>http://www.cio.gv.at</Adresse>  
</InternetAdresse>
```

Beispiel 16: Die Internetadresse www.cio.gv.at

```
<InternetAdresse>  
  <Adresse>webmaster@cio.gv.at</Adresse>  
</InternetAdresse>
```

Beispiel 17: Die Internetadresse kann auch eine e-mail Adresse sein

2.1.10 Telefonadressen

Englische Version: TelephoneAddress

Deutsche Version: TelefonAdresse

Beschreibung: Container für alle Arten von Telefonnummern

Parent Elemente (en, de): PersonData, PersonenDaten (siehe Abschnitt 2.1.1)

155

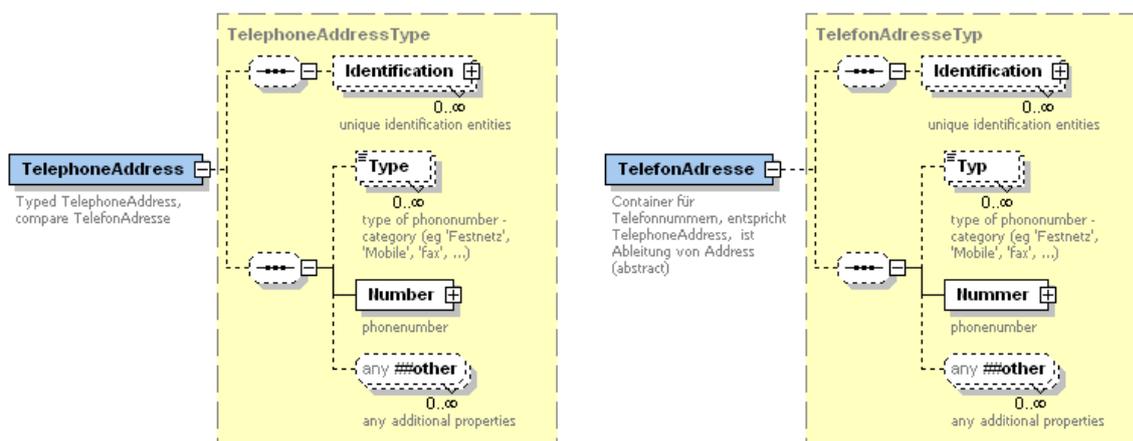


Abbildung 2.16: Unterlemente des TelephoneAddress und TelefonAdresse Containers

Englisch	Deutsch	Erklärung
Identification	Identifikation	Container für diverse Identifikationsmerkmale. (siehe Abschnitt 2.1.11)
Type	Typ	Typ der Telefonnummer, z.B. Festnetz, Mobil, Fax,...
Number	Nummer	Strukturierte Telefonnummer (siehe Unterpunkt 2.1.10)
##other	##other	Möglichkeit weitere Eigenschaften einzufügen, die bisher nicht angegeben wurden.

Tabelle 2.17: Unterelemente von TelephoneAddress und TelefonAdresse Elementen

Nummer

Englische Version: Number

Deutsche Version: Nummer

Beschreibung: Container für die genaue Angabe der Telefonnummer.

Parent Elemente (en, de): TelephoneAddress, TelefonAdresse (siehe Abschnitt 2.1.10)

160

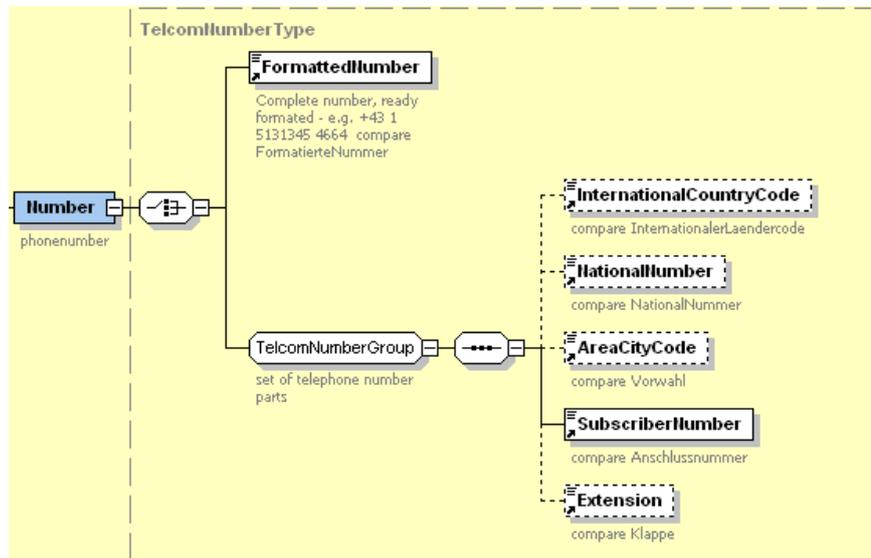


Abbildung 2.17: Unterlemente des Number Containers

Englisch	Deutsch	Erklärung
FormattedNumber	FormatierteNummer	Formatierte Nummer, z.B. +43 1 53134 4146
International- CountryCode	Internationaler- Laendercode	Internationaler Ländercode
NationalNumber	Nationalnummer	Manche Orte haben zwei Vorwahlen: für Gespräche aus In- und Ausland, diese ist die Inlandsvorwahl
AreaCityCode	Vorwahl	Vorwahl
SubscriberNumber	Anschlussnummer	Anschlussnummer
Extension	Klappe	Die Klappe (Durchwahl) der Nummer.

Tabelle 2.18: Unterlemente von Number und Nummer Elementen

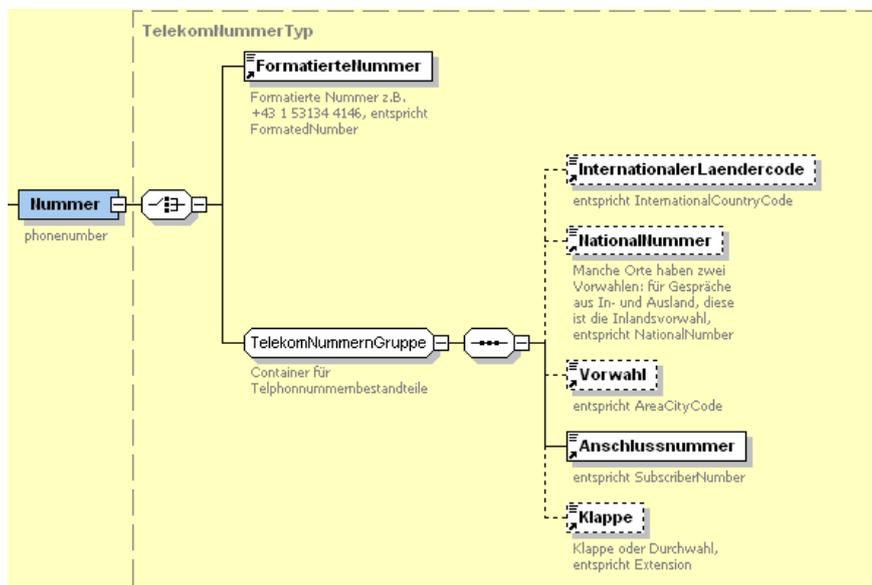


Abbildung 2.18: Unterlemente des Nummer Containers

Beispiele

```
<TelefonAdresse>
  <Typ>Festnetz</Typ>
  <Nummer>
    <FormatierteNummer>+43 1 53134 4146</FormatierteNummer>
  </Nummer>
</TelefonAdresse>
```

Beispiel 18: Eine einfache Telefonnummer im Festnetz

```
<TelefonAdresse>
  <Typ>Festnetz</Typ>
  <Nummer>
    <InternationalerLaendercode>43</InternationalerLaendercode>
    <Vorwahl>1</Vorwahl>
    <Anschlussnummer>53134</Anschlussnummer>
    <Klappe>4146</Klappe>
  </Nummer>
</TelefonAdresse>
```

Beispiel 19: Eine einfache Telefonnummer im Festnetz mit genauer Auflistung der Bestandteile

2.1.11 Identifikationsdaten

Englische Version: Identification

Deutsche Version: Identifikation

165

Beschreibung: Container für diverse Identifikationsmerkmale, besteht aus Type und Value Unterelementen, aus technischen Gründen nur auf Englisch verfügbar.

Parentelemente: Identifiaktionsdaten kommen in PersonData, PersonenDaten, PhysicalPerson, NatuerlichePerson, CorporateBodyund NichtNatuerlichePersonElementen vor.

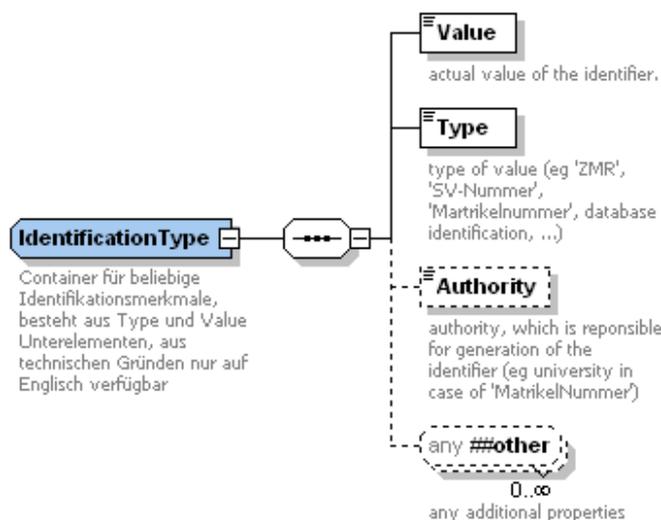


Abbildung 2.19: Unterelemente des Identification und des Identifikation Containers

Englisch	Deutsch	Erklärung
Value		Wert der Identifikation
Type		Attribute: Id Typ des Wertes, z.B. ZMR, SV-Nummer, Matrikelnummer,...
Authority		Dienststelle, welche für die Erstellung der Identifikation zuständig ist, z.B. Universität für eine Matrikelnummer
##other	##other	Möglichkeit weitere Eigenschaften einzufügen, die bisher nicht angegeben wurden.

Tabelle 2.19: Unterelemente von Identification und Identifikation Elementen

Beispiele

170

```
<Identification>
  <Value>0212123</Value>
  <Type>Matrikelnummer</Type>
  <Authority>TU-Wien</Authority>
</Identification>
```

Beispiel 20: Matrikelnummer der TU-Wien als Identifikation

```
<Identification>
  <Value>0102061134</Value>
  <Type>SV-Nummer</Type>
  <Authority>Sozialversicherungsanstalt der gewerblichen Wirtschaft</Authority>
</Identification>
```

Beispiel 21: SV-Nummer der Sozialversicherungsanstalt der gewerblichen Wirtschaft als Identifikation

2.1.12 Einfache Identifikationsdaten

Englische Version: AbstractSimpleIdentification

Deutsche Version: AbstractSimpleIdentification

Beschreibung: Sammelbegriff für verschiedene in Österreich häufig vorkommende Identifikationsnummern.

175

Parentelemente: Einfache Identifikationsdaten können als alternative zu Identifikation Elementen in PersonData, PersonenDaten, PhysicalPerson, NatuerlichePerson, CorporateBodyund NichtNatuerlichePersonElementen vorkommen.

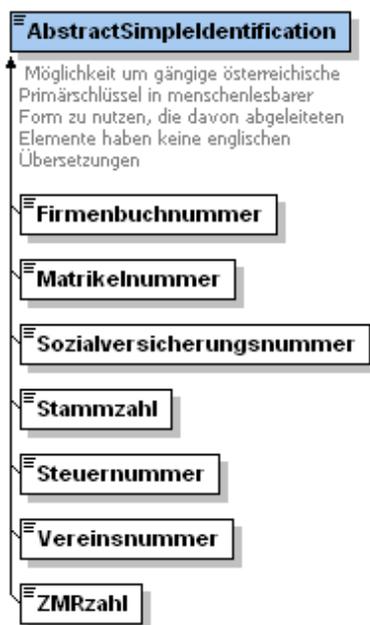


Abbildung 2.20: Unterlemente des AbstractSimpleIdentification Containers

Englisch	Deutsch	Erklärung
	Ergaenzungsregisterzahl	Ergänzungsregister für nicht-natürliche Personen
	Firmenbuchnummer	Firmenbuchnummer
	Matrikelnummer	Matrikelnummer
	Sozialversicherungsnummer	Sozialversicherungsnummer
	Stammzahl	Stammzahl
	Steuernummer	Steuernummer
	Vereinsnummer	Vereinsnummer
	ZMRzahl	ZMRzahl

Tabelle 2.20: Unterlemente von AbstractSimpleIdentification Elementen

Beispiele

`<Matrikelnummer>0212123</Matrikelnummer>`

Beispiel 22: Matrikelnummer

`<Sozialversicherungsnummer>0102061134</Sozialversicherungsnummer>`

Beispiel 23: Sozialversicherungsnummer

2.2 Alternative Root Elemente

180

Englische Version: Name

Deutsche Version: PersonenName

Beschreibung: Container für diverse Bestandteile eines Personennamens

Parent Elemente (en, de): PhysicalPerson, NatuerlichePerson (siehe Abschnitt 2.1.2)

2.2.1 Kompakte Personendaten

185

Englische Version: CompactPersonData

Deutsche Version: KompaktePersonenDaten

Beschreibung: Container für diverse kompakte Informationen über Personen

Kein Elternelement

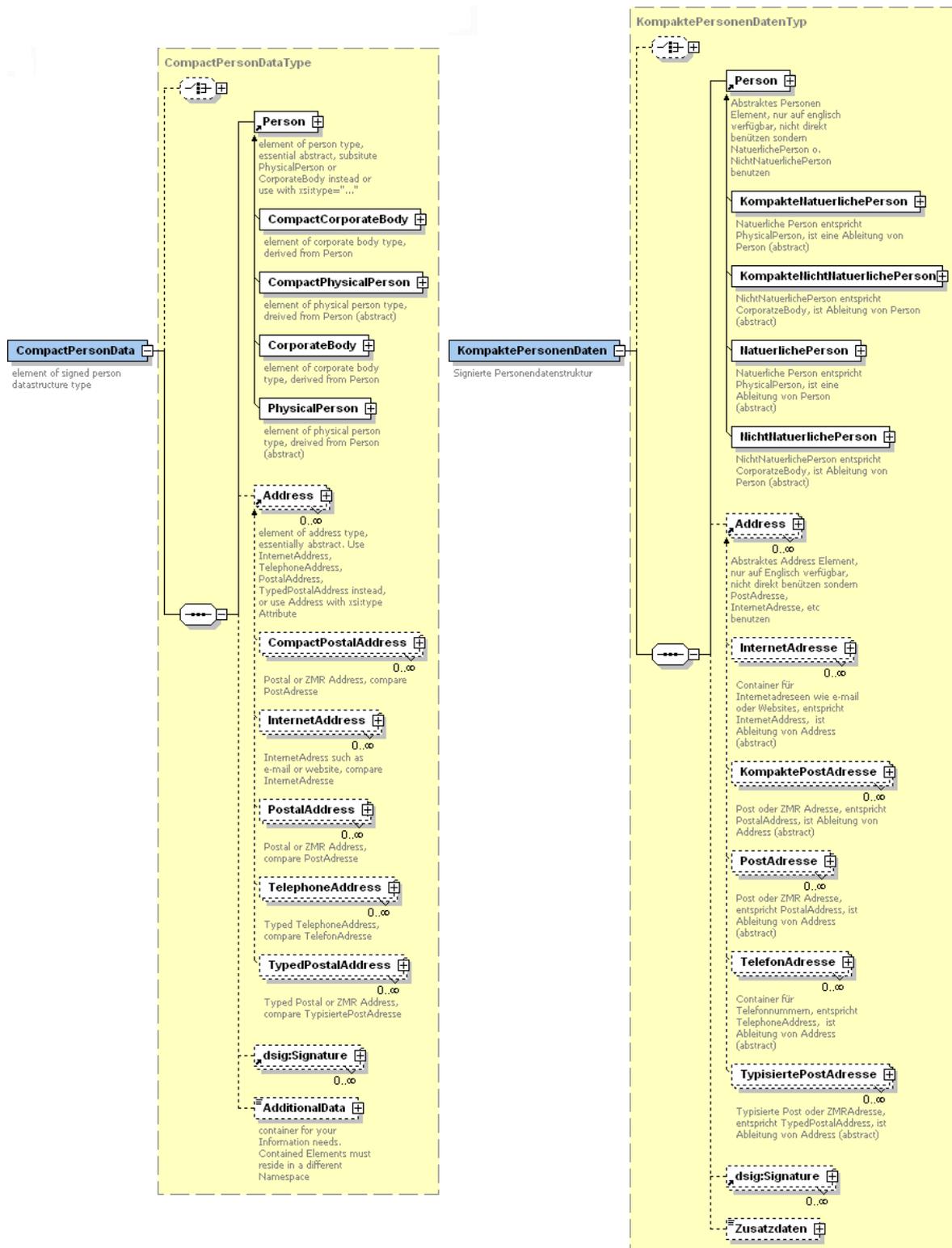


Abbildung 2.21: Unterlemente des CompactPersonData und KompaktePersonenDaten Elements

Englisch	Deutsch	Erklärung
Identification	Identifikation	Container für diverse Identifikationsmerkmale. (siehe Abschnitt 2.1.11)
AbstractSimple- Identification	AbstractSimple- Identification	Sammelbegriff für verschiedene in Österreich häufig vorkommende Identifikationsnummern. (siehe Abschnitt 2.1.12)
Person	Person	Abstraktes Personen Element, nur auf Englisch verfügbar, nicht direkt benützen, sondern NatuerlichePerson oder NichtNatuerlichePerson benutzen (siehe Abschnitt 2.2.2)
CompactPhysical- Person	Kompakte- NatuerlichePerson	(siehe Abschnitt 2.2.3)
CompactCorporate- Body	KompakteNicht- NatuerlichePerson	(siehe Abschnitt 2.1.2)
PhysicalPerson	NatuerlichePerson	(siehe Abschnitt 2.1.4)
CorporateBody	NichtNatuerliche- Person	
Address	Address	Abstraktes Address Element, nur auf Englisch verfügbar, nicht direkt benützen sondern PostAdresse, InternetAdresse, etc. benutzen (siehe Abschnitt 2.1.9)
InternetAddress	InternetAdresse	(siehe Abschnitt 2.2.5)
CompactPostal- Address	KompaktePost- Adresse	(siehe Abschnitt 2.1.7)
PostalAddress	PostAdresse	(siehe Abschnitt 2.1.10)
TelephoneAddress	TelefonAdresse	(siehe Abschnitt 2.1.8)
TypedPostalAddress	TypisiertePost- Adresse	
dsig:Signature	dsig:Signature	Internationaler Standard für digitale Signaturen, nur von technischem Interesse
AdditionalData	Zusatzdaten	Möglichkeit weitere, nicht vordefinierte, Daten einzugeben

Tabelle 2.21: Unterelemente von CompactPersonData und KompaktePersonenDaten Elementen

2.2.2 Kompakte natürliche Person

190

Englische Version: CompactPhysicalPerson

Deutsche Version: KompakteNatuerlichePerson

Beschreibung: Alternative, kompakte Form der NatuerlichenPerson (siehe Abschnitt 2.1.2)

Parent Elemente (en, de): PersonData, PersonenDaten (siehe Abschnitt 2.1.1)

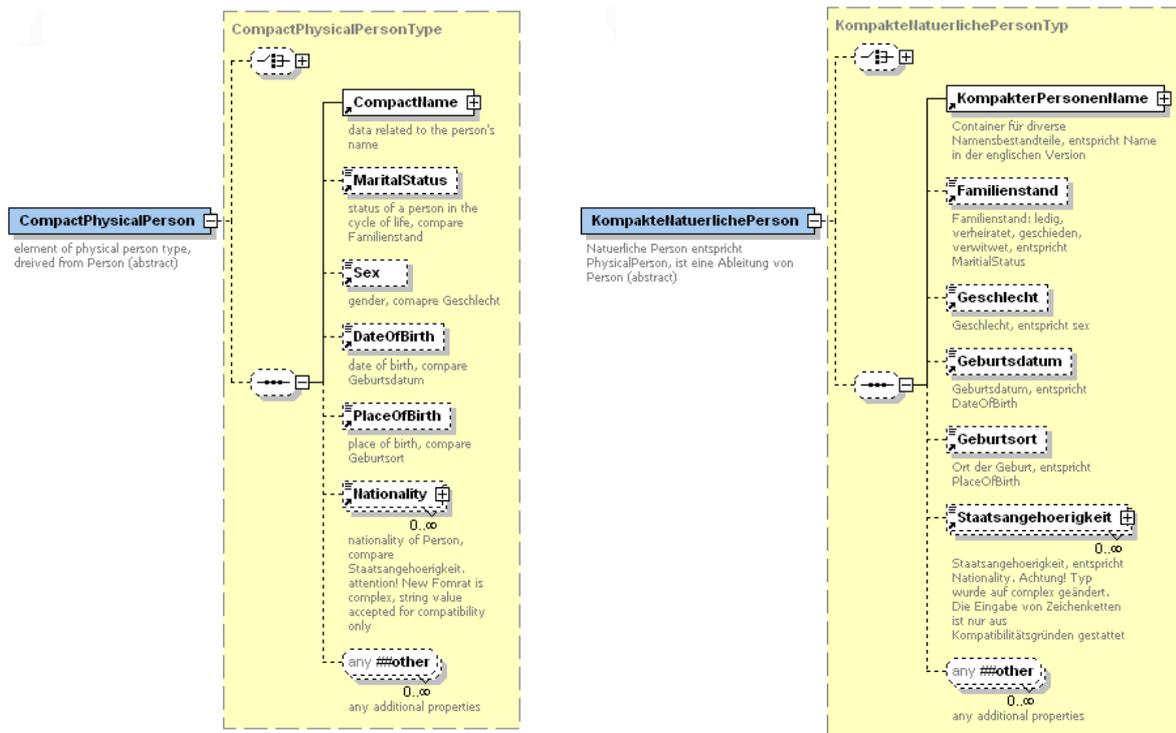


Abbildung 2.22: Unterlemente des CompactPhysicalPerson und KompakteNatuerlichePerson Elements

Englisch	Deutsch	Erklärung
Identification	Identifikation	Container für diverse Identifikationsmerkmale. (siehe Abschnitt 2.1.11)
AbstractSimple- Identification	AbstractSimple- Identification	Sammelbegriff für verschiedene in Österreich häufig vorkommende Identifikationsnummern. (siehe Abschnitt 2.1.12)
CompactName	KompakterPersonenName	Kompakter Container für diverse Namensbestandteile. (siehe Abschnitt 2.2.4)
MartialStatus	Familienstand	Mögliche Werte: (single/ledig, married/verheiratet, divorced/geschieden, widowed/verwitwet)
Sex	Geschlecht	Mögliche Werte: (male/männlich, female/weiblich, unknown/unbekannt)
DateOfBirth	Geburtsdatum	Geburtsdatum: Tag und/oder Monat können ausgelassen werden.
PlaceOfBirth	Geburtsort	Einfache Angabe des Geburtsortes.
Nationality	Staatsangehoerigkeit	(siehe Unterpunkt 2.1.2)
##other	##other	Möglichkeit weitere Eigenschaften einzufügen, die bisher nicht angegeben wurden.

Tabelle 2.22: Unterelemente von CompactPhysicalPerson und KompakteNatuerlichePerson Elementen

Beispiele

195

```
<KompakteNatuerlichePerson>
  <KompakterPersonenName>
    <Vorname>Max</Vorname>
    <Familiennamen>Mustermann</Familiennamen>
  </KompakterPersonenName>
</KompakteNatuerlichePerson>
```

Beispiel 24: Max Mustermann - der Name ist die kompakteste Form für natuerliche Personen

```
<KompakteNatuerlichePerson>
  <KompakterPersonenName>
    <Vorname>Max</Vorname>
    <Familiennamen>Mustermann</Familiennamen>
  </KompakterPersonenName>
  <Familienstand>ledig</Familienstand>
  <Geschlecht>männlich</Geschlecht>
</KompakteNatuerlichePerson>
```

Beispiel 25: Herr Max Mustermann ist ledig

2.2.3 Kompakte nicht natürliche Person

Englische Version: CompactCorporateBody

Deutsche Version: KompakteNichtNatuerlichePerson

Beschreibung: Alternative, kompakte Form der NichtNatuerlichenPerson (siehe Abschnitt 2.1.4)

200

Parent Elemente (en, de): PersonData, PersonenDaten (siehe Abschnitt 2.1.1)

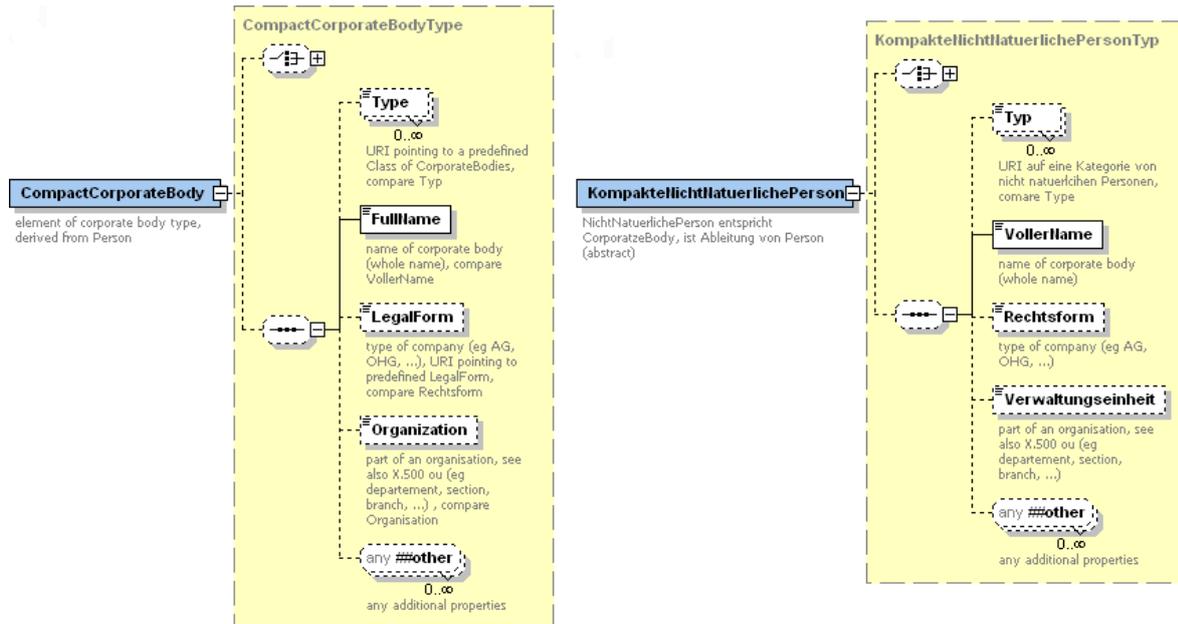


Abbildung 2.23: Unterlemente des CompactCorporateBody und KompakteNichtNatuerlichePerson Containers

Englisch	Deutsch	Erklärung
Typ	Typ	URI auf eine Kategorie von nicht natürlichen Personen
FullName	VollerName	Voller Name der nicht natürlichen Person
LegalForm	Rechtsform	Rechtsform der Firma, z.B. AG, OGH, ...
Organization	Verwaltungseinheit	Teil der Firma wie Sektion, Abteilung,...
##other	##other	Möglichkeit weitere Eigenschaften einzufügen, die bisher nicht angegeben wurden.

Tabelle 2.23: Unterelemente von CompactCorporateBody und KompakteNichtNaturerlichePersonElementen

Beispiele

```
<KompakteNichtNatuerlichePerson>  
  <VollerName>Druckerei Ganster</VollerName>  
</KompakteNichtNatuerlichePerson>
```

Beispiel 26: Die Druckerei Ganster

```
<KompakteNichtNatuerlichePerson>  
  <VollerName>Druckerei Ganster</VollerName>  
  <Rechtsform>AG</Rechtsform>  
</KompakteNichtNatuerlichePerson>
```

Beispiel 27: Die Druckerei Ganster mit Zusatzinformation der Rechtsform

2.2.4 Kompakter Personennamen

Englische Version: CompactName

Deutsche Version: KompakterPersonenName

Beschreibung: Alternative, kompakte Form des Personennamens (siehe Abschnitt 2.1.3)

Parent Elemente (en, de): PersonData, PersonenDaten (siehe Abschnitt 2.1.1)

205

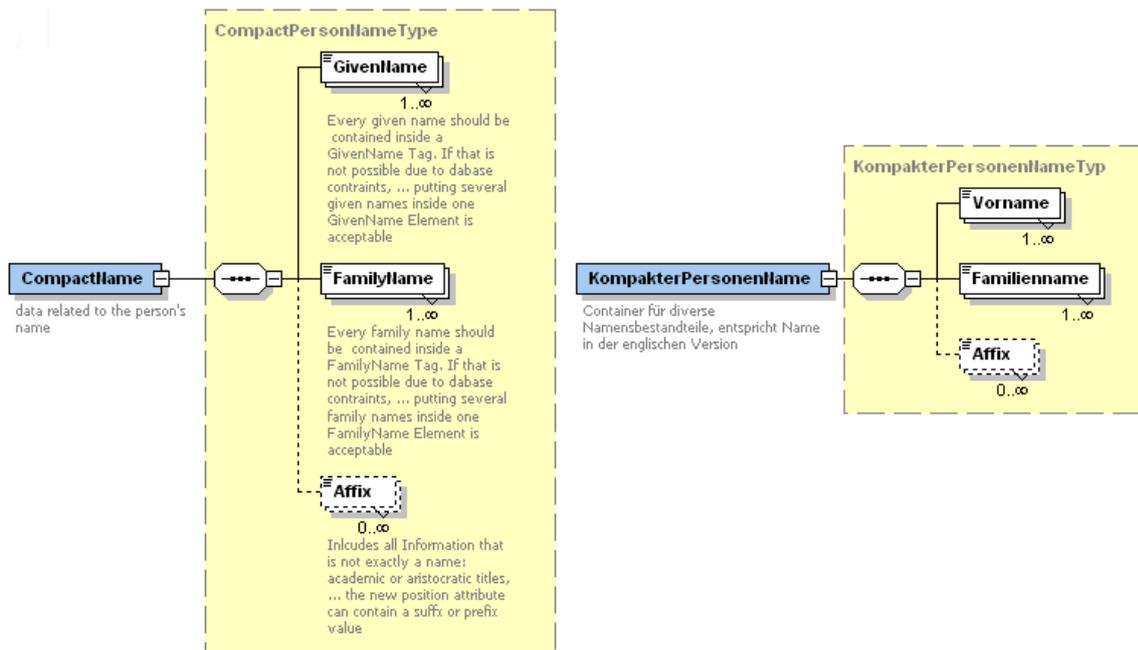


Abbildung 2.24: Unterlemente des CompactName und KompakterPersonenName Containers

Englisch	Deutsch	Erklärung
GivenName	Vorname	Vorname, Jeden Vornamen in eigenes GivenName Tag einschließen
FamilyName	Familiennamen	Familiennamen, jeden Familiennamen der durch ein Leerzeichen getrennt ist, in einen eigenes FamilyName Tag einschließen. Attribute: primary/primär (true/wahr, false/falsch, undefined/undefiniert) prefix
Affix	Affix	Pre- oder Postfix: z.B: akademische Titel, ... Attribute: type/typ (academicGrade/akademischerGrad, aristocraticPrefix/–, aristocraticTitle/Adelstitel, familyNamePrefix/FamiliennamenAffix, familyNameSuffix/–, –/Anrede, formOfAddress/– generation/Generation, qualification/Qualifikation) position (prefix, suffix)

Tabelle 2.24: Unterelemente von CompactName und KompakterPersonenNameElementen

Beispiele

```
<KompakterPersonenName>  
  <Vorname>Emilie</Vorname>  
  <Familiennamenname>Mustermann</Familiennamenname>  
</KompakterPersonenName>
```

Beispiel 28: Emilie Mustermann

```
<KompakterPersonenName>  
  <Vorname>Alexander</Vorname>  
  <Familiennamenname>Leiningen-Westerburg</Familiennamenname>  
  <Affix type="akademischerGrad" position="prefix">Mag.</Affix>  
  <Affix type="akademischerGrad" position="suffix">MAS</Affix>  
</KompakterPersonenName>
```

Beispiel 29: Mag. Alexander Leiningen-Westerburg, MAS

2.2.5 Kompakte Postadresse

Englische Version: CompactPostalAddress

210

Deutsche Version: KompaktePostAdresse

Beschreibung: Alternative, kompakte Form des Personennamens (siehe Abschnitt 2.1.7)

Parent Elemente (en, de): PersonData, PersonenDaten (siehe Abschnitt 2.1.1)

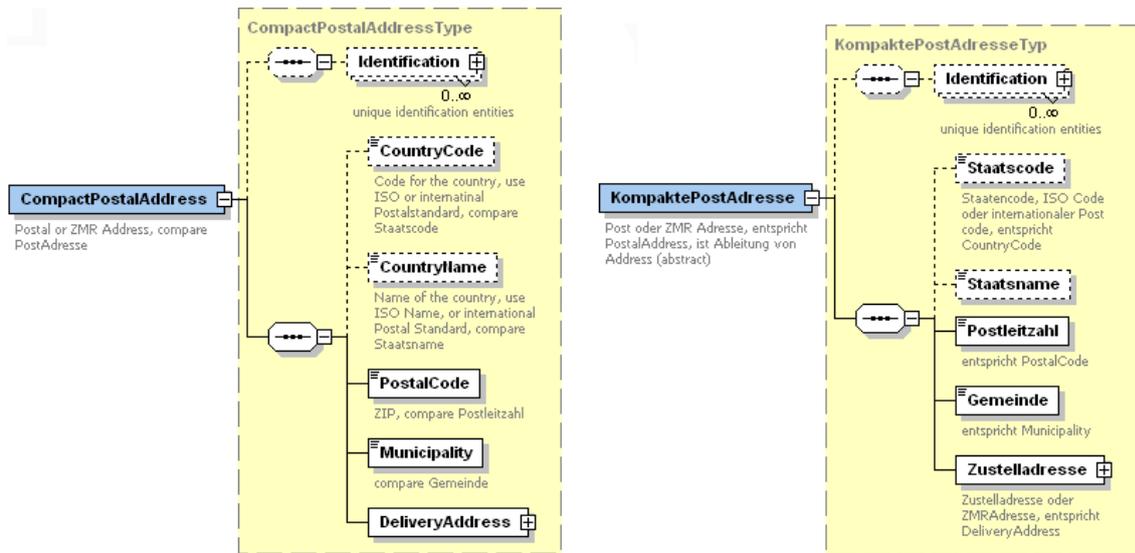


Abbildung 2.25: Unterlemente des CompactPostalAddress und KompaktePostAdresse Containers

Englisch	Deutsch	Erklärung
CountryCode	Staatscode	Staatencode, ISO Code oder internationaler Postcode
CountryName	Staatsname	Staatsname, ISO Namen benutzen.
PostalCode	Postleitzahl	Postleitzahl
Municipality	Gemeinde	Gemeinde
DeliveryAddress	Zustelladresse	Zustelladresse (siehe Unterpunkt 2.2.5)

Tabelle 2.25: Unterlemente von CompactPostalAddress und KompaktePostAdresseElementen

Zustelladressen

Englische Version: DeliveryAddress

215

Deutsche Version: Zustelladresse

Beschreibung: Alternative, kompakte Form des Personennamens (siehe Abschnitt ??)

Parent Elemente (en, de): CompactPostalAddress, KompaktePostAdresse (siehe Abschnitt 2.2.5)

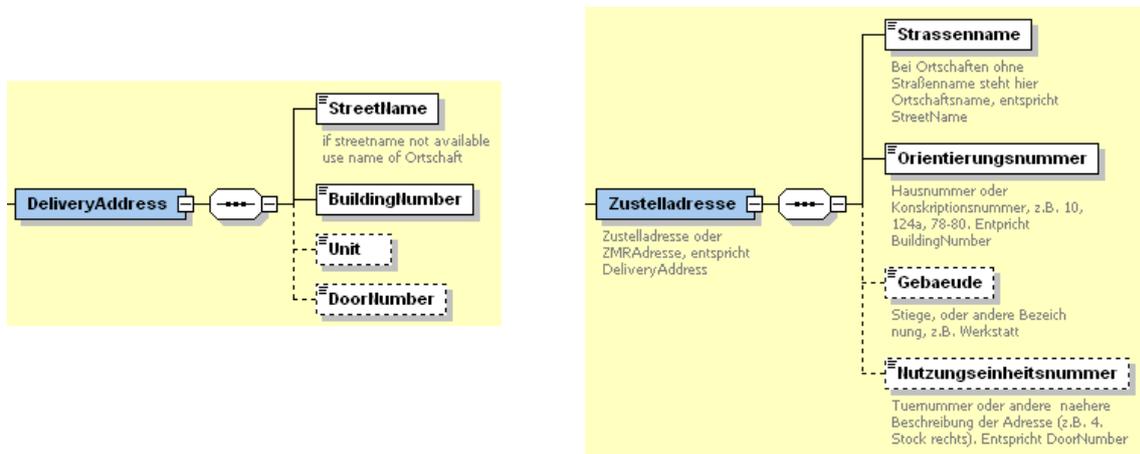


Abbildung 2.26: Unterlemente des DeliveryAddress und Zustelladresse Containers

Englisch	Deutsch	Erklärung
StreetName	Strassenname	Bei Ortschaften ohne Straßennamen steht hier Ortschaftsname
BuildingNumber	Orientierungsnummer	Hausnummer oder Konstruktionsnummer, z.B. 10, 124a, 78–80
Unit	Gebaeude	Gebäudeteil oder eigenes Gebäude, z.B. Stiege, Werkstatt,...
DoorNumber	Nutzungseinheit	Tuernummer oder andere naechere Beschreibung der Adresse, z.B. 4. Stock rechts

Tabelle 2.26: Unterelemente von BankConnection und Bankverbindung Elementen

Beispiele

220

```
<KompaktePostAdresse type="Anschrift">
  <Postleitzahl>1010</Postleitzahl>
  <Gemeinde>Wien</Gemeinde>
  <Zustelladresse>
    <Strassenname>Ringstrasse</Strassenname>
    <Orientierungsnummer>1</Orientierungsnummer>
  </Zustelladresse>
</KompaktePostAdresse>
```

Beispiel 30: Die Postanschrift 1010 Wien, Ringstraße 1 in kompakter Form

```
<KompaktePostAdresse type="Anschrift">
  <Staatsname>Österreich</Staatsname>
  <Postleitzahl>1010</Postleitzahl>
  <Gemeinde>Wien</Gemeinde>
  <Zustelladresse>
    <Strassenname>Ringstraße</Strassenname>
    <Orientierungsnummer>1</Orientierungsnummer>
    <Gebaeude>2. Stiege</Gebaeude>
    <Nutzungseinheit>4. Stock rechts</Nutzungseinheit>
  </Zustelladresse>
</KompaktePostAdresse>
```

Beispiel 31: Die Postanschrift 1010 Wien, Ringstraße 1 in kompakter Form mit näheren Erklärungen

Kapitel 3

PersonData 2.0 für Entwickler/Programmierer

praxis beispiele mit “blaupausenansatz” (mypersondata), compactpersondata, ws ansatz, entwicklung eigener spezifikationen, ...

225

3.1 Voraussetzungen

Tiefes Verständnis der `PersonData` Struktur setzt XML Schema Kenntnisse voraus, die das überliche Einsteigermaß (Wie baue ich ein Schema für `<Book>?`) deutlich übersteigen. Z.B. sollten Sie mit abstrakten Typen und abstrakten Elementen vertraut sein. Zur Vorbereitung empfiehlt sich das W3C Schema Tutorial xxx oder klassische Sekundärliteratur (z.B. xxx Oreilly).

230

3.2 Technische Änderungen gegenüber Version 1.0

Die augenfälligsten Veränderungen gegenüber Version 1.0 sind die deutschsprachige Version und neue Elemente, z.B im Bereich Bankverbindung und Identifikation. Ebenfalls neu ist die Zusammenführung der `PersonData` Struktur in ein einzelnes Schema, weil das rekursive Einbinden von Schemateilen bei manchen Parsern zu Fehlverhalten führte.

235

Auf technische Seite wurde die Substitution Group neu eingeführt. In Version 1.0 war es notwendig die von den abstrakten Typen `AbstractPersonType` und `AbstractAddressType` abgeleiteten Elemente `Person` und `Address` in der XML Instanz durch konkrete Elemente zu überlagern:

240

```
<Person xsi:type="PhysicalPerson">...</Person>  
<Address xsi:type="PostalAddress">...</Address>
```

In Version 2.0 wurden die konkreten Elemente `PhysicalPerson`, `PostalAddress`, ... der Substitution Group `Person` bzw. `Address` hinzugefügt und sie können jetzt direkt in der Instanz benutzt werden.

245

```
<PhysicalPerson>...</PhysicalPerson>
<PostalAddress>...</PostalAddress>
```

Im Gegenzug fiel die `abstract` Deklaration der Typen `AbstractPersonType` und `AbstractAddressType`, weil selbige nicht kompatibel mit Substitution Groups ist. Ebenso nicht vorhanden ist, eine `abstract` Deklaration der Elemente `Person` und `Address`, die man beim Gebrauch einer Substitution Group erwarten würde, weil sie nicht rückwärtskompatibel zu Version 1.0 wäre. 250

3.3 “Blaupausenkonzert” – myPersonData

Eine komplette PersonData Struktur verfügt für die meisten Anwendungen über zu viele Elemente. Auch die vereinfachten Elemente z.B. `CompactPersonData` erfüllen manchmal nicht alle Ansprüche. Um in einem WebService Umfeld eine genau passende PersonData Struktur benutzen zu können, ist es notwendig sich seine eigene Version einer PersonData Struktur abzuleiten. Die PersonData Struktur fungiert dabei als Blaupause. Alle Elemente die in der PersonData Struktur als optional gekennzeichnet sind können in der abgeleiteten Struktur gelöscht werden. Steht in der PersonData Struktur an einer Stelle ”##other”, so können an dieser Stelle beliebige XML Elemente aus einem anderen Namespace eingebaut werden. 255

Beim Blaupausenkonzert ist es prinzipiell auch möglich auf die Ableitung von abstrakten Basis Typen wie z.B. dem `Person` Element, oder dem `Address` Element zu verzichten. Statt dessen kann man z.B. eine Abfolge von `PostalAddress` und `InternetAddress` fordern. 260

Wichtig beim Blaupausenkonzert ist es, dass XML-Schemata, die aus dem resultierenden Schema abgeleitet wurden, dem original PersonData Schema genügen. Der Namespace des abgeleiteten Schemas entspricht dem der originalen PersonData Struktur. Die Abbildung 3.1 zeigt eine sehr einfache abgeleitete Struktur (Beispielbeschreibung 3.3.2). Trotzdem sind alle XML Dokumente die diesem Schema gehorchen automatisch auch PersonData konform. 265

3.3.1 Vorgehensweise

Die Vorgehensweise für das erstellen eigener Schemata sieht wie folgt aus.

- Feststellen, welche Daten benötigt werden.
- Ein kleines XML-Testfile gestalten
- Anstatt ##other einen eigenen Namespace generieren 275
- Dieses File gegen das originale PersonData Schema validieren
- Die nicht benötigten Elemente löschen (nur bei optionalen Elementen möglich)
- Das XML-File gegen das neue Schema validieren
- Elemente verpflichtend (compulsary) machen.
- Substitues erstellen (choice of legacy) 280

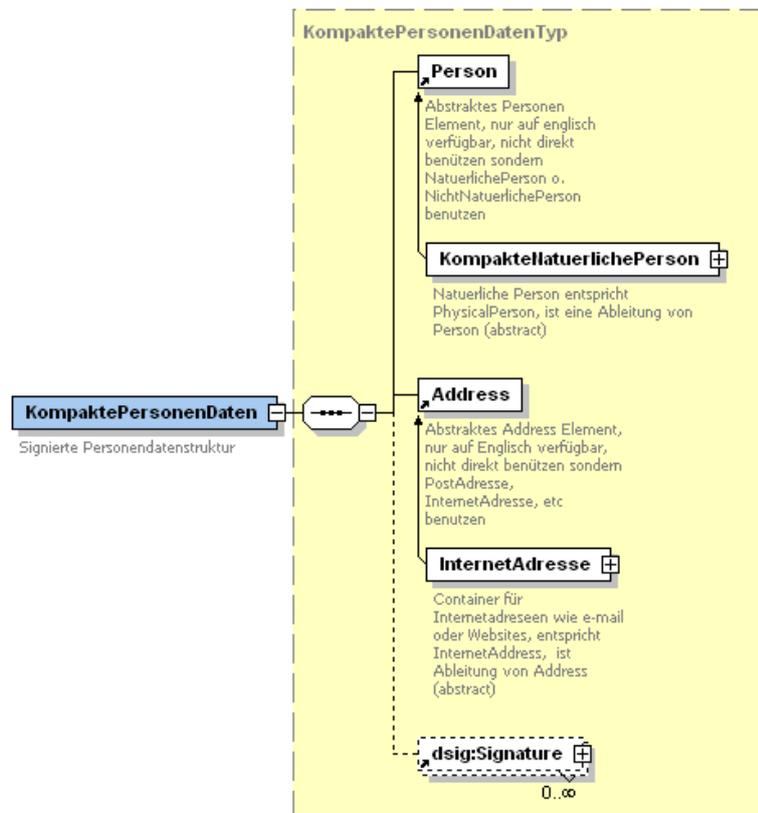


Abbildung 3.1: Einfaches Schema, Unterelemente des Root-Elements

- Das XML-File gegen das aktuelle Schema validieren

Im folgenden werden wir einige Beispiele betrachten, um diese Schritte anschaulicher zu machen. Im Anschluss wird eine genaue Erklärung der einzelnen Schritte präsentiert.

3.3.2 Beispiel 1

Als einführendes Beispiel werden wir die Entwicklung eines Schemas betrachten, das ausschließlich den Namen und die dazugehörige E-Mail definiert. Dies kann z.B. für Netzdienste von Nutzen sein, die Newsletter anbieten und zur Anmeldung nicht mehr Personendaten benötigen, als den Namen und die E-Mail Adresse. 285

Zugegebendermaßen ist dieses Beispiel nicht gerade praxisnahe, da bei Newsletteranmeldungen und Übertragungen wohl einfachere Hilfsmittel verwendet werden als XML Dateien und Schematas. 290

- Wie schon erwähnt werden für unseren Netzdienst nur der Personennamen und die E-Mail Adresse benötigt. Am einfachsten beginnt man, indem man das vorhandene Schema unter einem neuen Namen speichert (z.B. "MyPersonData_newsletter.xsd"). 295

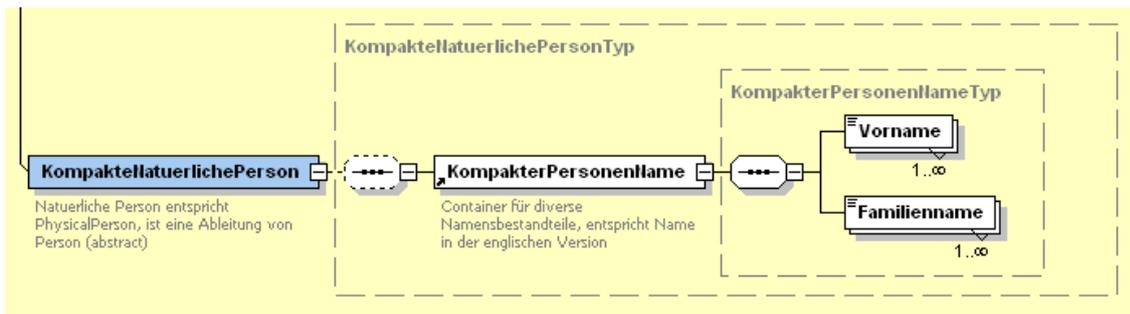


Abbildung 3.2: Einfaches Schema, Unterelemente des KompakteNatuerlichePerson-Elements

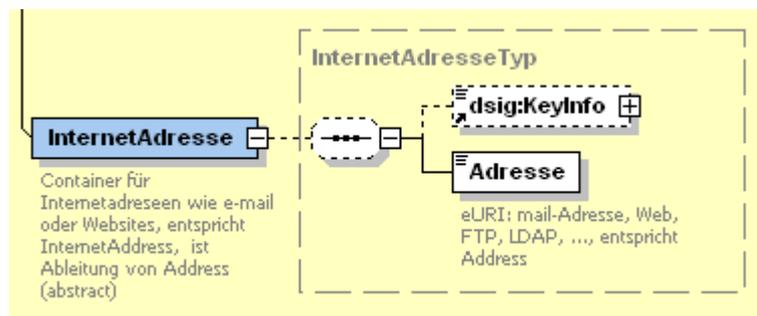


Abbildung 3.3: Einfaches Schema, Unterelemente des InternetAdresse-Elements

- Nun generieren wir eine XML-Beispieldatei, die so einige Möglichkeiten der zukünftigen Nutzung beinhalten sollte. In unserem Beispiel ist diese Datei recht einfach (Beispiel 3.3.2).

```

<?xml version="1.0" encoding="UTF-8"?>
<KompaktePersonenDaten
  xmlns="http://reference.e-government.gv.at/namespace/
                                persondata/de/20040201#"
  xmlns:dsig="http://www.w3.org/2000/09/xmldsig#"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://reference.e-government.gv.at/namespace/
                    MyPersonData_newsletter.xsd">
  <KompakteNatuerlichePerson>
    <KompakterPersonenName>
      <Vorname>Max</Vorname>
      <Familiennamen>Mustermann</Familiennamen>
    </KompakterPersonenName>
  </KompakteNatuerlichePerson>
  <InternetAdresse>
    <Adresse>max.mustermann@mail.at</Adresse>
  </InternetAdresse>
</KompaktePersonenDaten>

```

Beispiel 32: Beispiel XML Datei

Das Wissen um die Bedeutung der obigen Zeilen darf vorausgesetzt werden.

- Im nächsten Schritt validieren wir diese XML Datei gegen das originale Schema. Bei 300
erfolgreicher Überprüfung beginnen wir alle Elemente die nicht benötigt werden zu
löschen (nur optionale Elemente).
*Nicht vergessen: nach jedem löschen das Schema selbst validieren um nicht wichtige
oder abhängige Elemente zu löschen.*
- Nach erfolgreichem Löschen ist schon das minimale Schema übrig. Die weitere Vorge- 305
hensweise besteht darin, dass wir alle Elemente, die im Schema optional sind und in
unserem aktuellen Fall Pflichtelemente sein sollen auch als solche zu deklarieren
(minOcc=1).

Nach jedem Schritt nicht vergessen: die XML Datei gegen das Schema validieren! 310

- Als *Ergebnis* erhalten wir jenes Schema, welches sich aus den Abbildungen 3.1, 3.2, 3.3
zusammensetzt.
Dieses Ergebnisschema ist das minimale Schema für unseren Einsatz, welches vom
originalen abgeleitet werden kann.

3.3.3 Beispiel 2

315

Als komplexes Beispiel betrachten wir einen Einsatz der Personendatenstruktur für den Datenaustausch von kompletten Firmendaten, welche den Firmennamen (inklusive aller

Zusätze), Adresse und Kontaktmöglichkeit beinhalten. Weiters wird ein eigener Namespace definiert, der noch zusätzlich eine Bewertung der Firma und die Speicherung eines Werbetextes über diese möglich macht. Das neue Element `<bewertung>` welche die Zahlen von 1-5 (Schulnotensystem) enthält wird in `NichtNatürlichePerson` hinzugefügt. `<werbung>` wird in `PersonenDaten` unter `Zusatzdaten` eingebaut. 320

- Analog zu Beispiel 1 3.3.2 erstellen wir nun zwei XML-Dateien (3.3.3 und 3.3.3), welche Beispieldaten enthalten. In unserem Fall enthält die erste Datei alle möglichen Elemente, die eine XML-Datei später enthalten kann, die zweite hingegen beinhaltet das minimale Vorkommen diverser Elemente. 325

```

<?xml version="1.0" encoding="UTF-8"?>
<PersonenDaten
  xmlns="http://reference.e-government.gv.at/namespace/persondata/de/20040201#"
  xmlns:mns="http://www.ownnamespace.at/mein_namespace"
  xmlns:dsig="http://www.w3.org/2000/09/xmlsig#"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="
    http://reference.e-government.gv.at/namespace/persondata/de/20040201#
    MyPersonData_2.xsd
    http://www.ownnamespace.at/mein_namespace
    MyNameSpace.xsd"
>
  <NichtNatuerlichePerson>
    <Firmenbuchnummer>123456</Firmenbuchnummer>
    <VollerName>Druckerei Ganster GesmbH</VollerName>
    <AlternativName>DruckGans</AlternativName>
    <AlternativName>Ganster Druck</AlternativName>
    <Rechtsform>GesmbH</Rechtsform>
    <Bankverbindung>
      <Inhaber>Druckerei Ganster GesmbH</Inhaber>
      <BankName>Die Erste</BankName>
      <NationaleBankverbindung>
        <Kontonummer>01234567890</Kontonummer>
        <BLZ>12345</BLZ>
      </NationaleBankverbindung>
    </Bankverbindung>
    <mns:bewertung>4</mns:bewertung>
  </NichtNatuerlichePerson>
  <PostAdresse>
    <Postleitzahl>1010</Postleitzahl>
    <Bundesland>Wien</Bundesland>
    <Gemeinde>Wien</Gemeinde>
    <Ortschaft>Wien</Ortschaft>
    <Zustelladresse>
      <Strassenname>Ringstraße</Strassenname>
      <Orientierungsnummer>1</Orientierungsnummer>
      <Nutzungseinheit>4. Stock, rechts</Nutzungseinheit>
    </Zustelladresse>
    <Empfaenger>
      <PersonenName>
        <Vorname>Max</Vorname>
        <Familiename>Mustermann</Familiename>
      </PersonenName>

```

```
<Zusatztext>Personalchef</Zusatztext>
<Verwaltungseinheit>Personalbüro</Verwaltungseinheit>
<Unternehmen>Druckerei Ganster GesmbH</Unternehmen>
</Empfaenger>
</PostAdresse>
<TelefonAdresse>
  <Typ>Festnetz</Typ>
  <Nummer>
    <InternationalerLaendercode>43</InternationalerLaendercode>
    <Vorwahl>1</Vorwahl>
    <Anschlussnummer>53134</Anschlussnummer>
    <Klappe>4146</Klappe>
  </Nummer>
</TelefonAdresse>
<InternetAdresse>
  <Adresse>www.druck-gans.at</Adresse>
</InternetAdresse>
<InternetAdresse>
  <Adresse>druck@druck-gans.at</Adresse>
</InternetAdresse>
<Zusatzdaten>
  <msn:werbung>Das ist ein Werbetext!</msn:werbung>
</Zusatzdaten>
</PersonenDaten>
```

Beispiel 34: Fortsetzung: Beispiel XML Datei für alle Elemente

```

<?xml version="1.0" encoding="UTF-8"?>
<PersonenDaten
  xmlns="http://reference.e-government.gv.at/namespace/persondata/de/20040201#"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="
    http://reference.e-government.gv.at/namespace/persondata/de/20040201#
    MyPersonData_2.xsd
    http://www.ownnamespace.at/mein_namespace
    MyNameSpace.xsd">
  <NichtNatuerlichePerson>
    <Firmenbuchnummer>123456</Firmenbuchnummer>
    <VollerName>Druckerei Ganster GesmbH</VollerName>
    <Rechtsform>GesmbH</Rechtsform>
    <mns:bewertung>4</mns:bewertung>
  </NichtNatuerlichePerson>
  <PostAdresse>
    <Postleitzahl>1010</Postleitzahl>
    <Bundesland>Wien</Bundesland>
    <Gemeinde>Wien</Gemeinde>
    <Ortschaft>Wien</Ortschaft>
    <Zustelladresse>
      <Postleitzahl>1234</Postleitzahl>
    </Zustelladresse>
  </PostAdresse>
  <TelefonAdresse>
    <Typ>Mobil</Typ>
    <Nummer>
      <FormatierteNummer>+43 (0) 664 123 45 67</FormatierteNummer>
    </Nummer>
  </TelefonAdresse>
  <Zusatzdaten>
    <msn:werbung>Das ist ein Werbetext!</msn:werbung>
  </Zusatzdaten>
</PersonenDaten>

```

Beispiel 35: Beispiel XML Datei für minimale Elemente

Wie man aus diesen Beispielen sieht wurden `bewertung` und `werbung` aus dem Namespace `http://www.ownnamespace.at/mein_namespace` (siehe Abschnitt 3.3.3) importiert. Bei `werbung` wäre aufgrund der `##any` definition zwar kein eigener Namespace nötig, aufgrund der genaueren Definition dieses Elements und eventueller Restriktionen ist es von Vorteil, auch `##any` Elemente im eigenen Namespace zu definieren. 330

- Jetzt werden wieder alle optionalen Elemente, die in unserem Schema nicht benötigt

werden gelöscht.

Nicht vergessen: nach jedem löschen das Schema selbst validieren um nicht wichtige oder abhängige Elemente zu löschen. 335

- Nach positiver validierung werden nun die Restriktionen geändert - in unserem Fall werden optionale Elemente verpflichtend gemacht. In einigen Fällen kann es nötig sein, weitere Restriktionen in Form von etwa definierten Zahlengrenzen bei der Postleitzahl einzuführen. 340
- Als Ergebnis kommt das Schema "MyPersonData_2.xsl" (siehe Abschnitt 3.3.3).

MyNameSpace Schema



Abbildung 3.4: MyNameSpace

In Abbildung 3.4 sehen sie die zwei globalen Elemente **bewertung** und **werbung**. Die Definition beider Elemente sehen sie in Tabelle 3.1. Aufgrund der rein deutschen Nutzung dieses Schemas gibt es keine englischen Pendants.

345

Deutsch	Erklärung
bewertung	Bewertung der Firma im Schulnotensystem. Falls keine Bewertung möglich ist kann man 0 Eintragen
werbung	Reines Textfeld zur Eingabe des Werbetextes

Tabelle 3.1: Globale Elemente von MyNameSpace

MyPersonData_2 Schema

Die Unterelemente von MyPersonData_2, NichtNatuerlichePerson, Postadresse, Zustelladresse, Empfängername und Personennamen sehen Sie in den Abbildungen. Die Elemente von Internetadresse, Telefonadresse und Zusatzdaten haben sich gegenüber der originalen Personendatenstruktur nicht verändert.

350

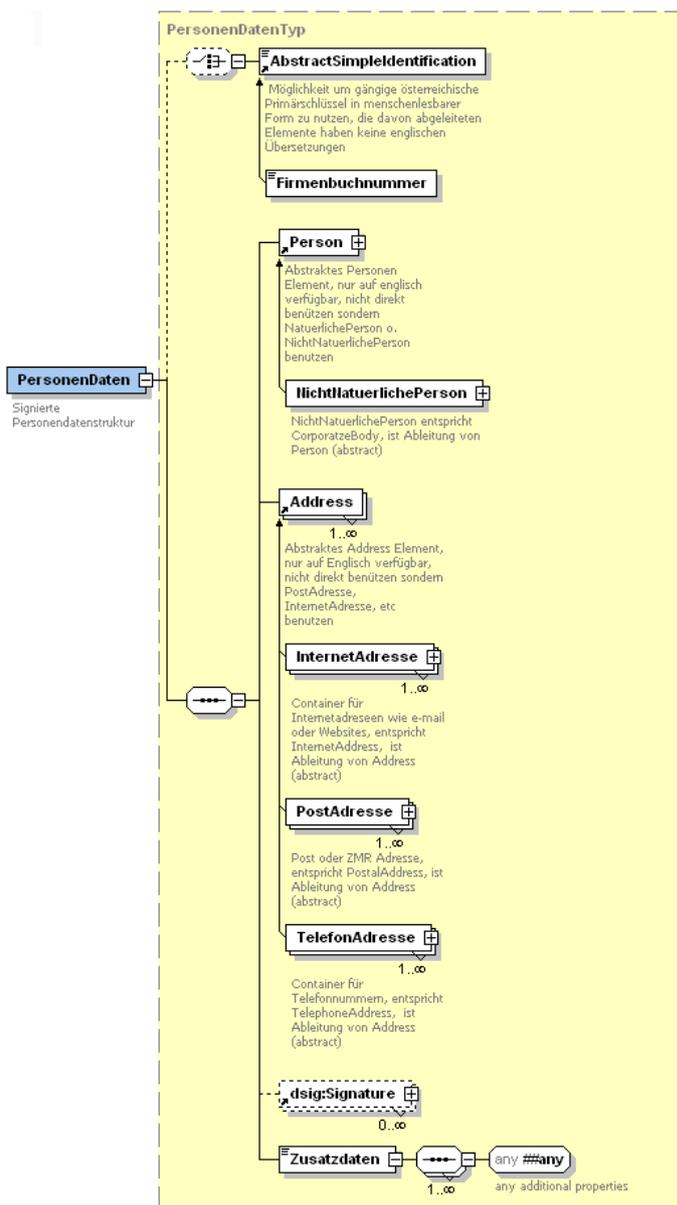


Abbildung 3.5: MyPersonData_2

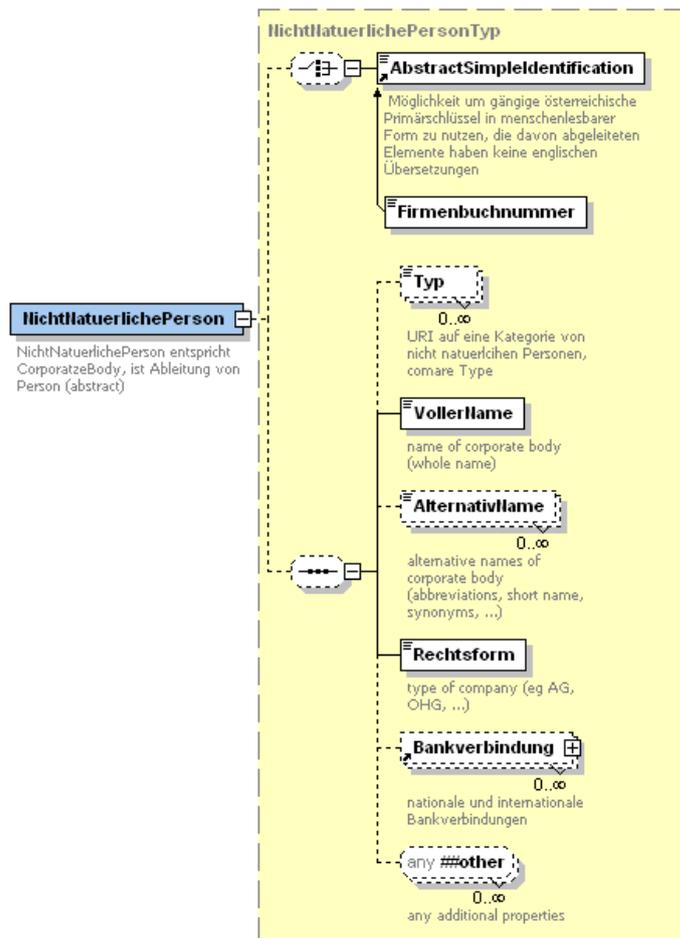


Abbildung 3.6: Unterelemente des NichtNatuerlichePerson Elements

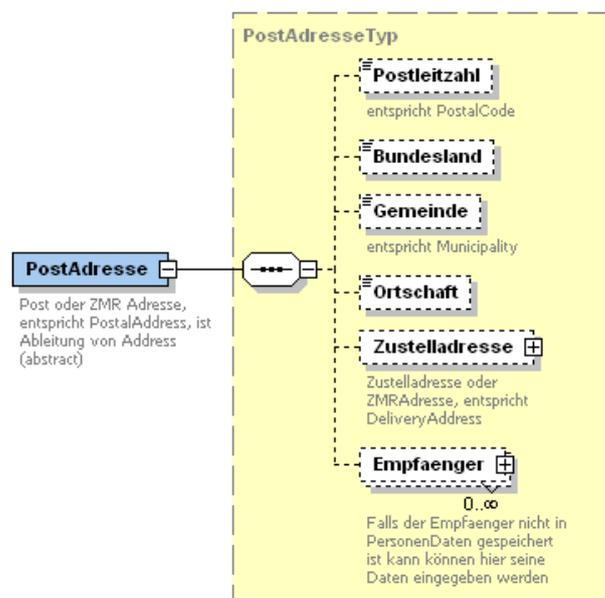


Abbildung 3.7: Unterelemente des Postadresse Elements

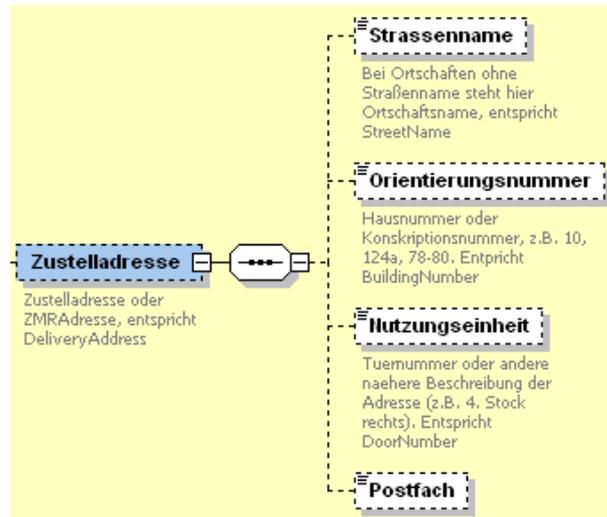


Abbildung 3.8: Unterelemente des Zustelladresse Elements

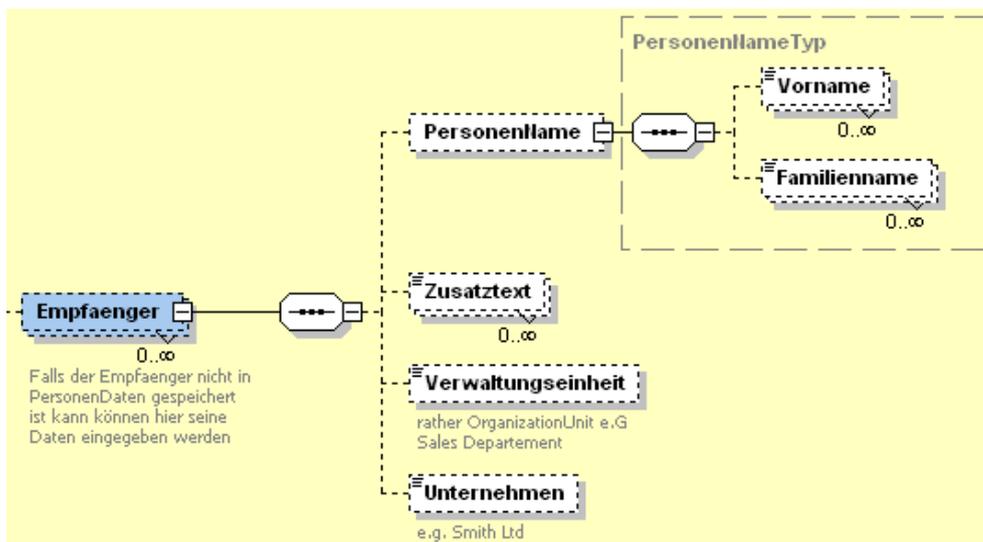


Abbildung 3.9: Unterelemente des Empfänger und Personenname Elements

3.4 CompactPersonData Ansatz

Der CompactPersonData Ansatz ist nur ein Spezialfall des Blaupausenansatzes, der auf vorgefertigte Vereinfachungen setzt. Für viele Anwendungen werden die Vereinfachungen relativ gut passen. Ist der Grad der Vereinfachung nicht ausreichend, oder fehlen Elemente, dann muß auf den Blaupauseansatz zurückgegriffen werden.

355

3.5 “Atomare Elemente” Ansatz

Wenn es unbedingt notwendig ist (Voraussetzungen siehe Abschnitt 3.7.1) können auch “Atomare Elemente” direkt angesprochen werden, wie man im XML-Beispiel 3.5 sehen kann. Diese Verwendung erfordert keine Neugestaltung der kompletten Struktur im Rahmen des Blaupausenkonzepts (siehe Abschnitt 3.3), sie bietet aber auch im folgenden nicht die Möglichkeit, zusätzliche Elemente oder Restriktionen einzufügen!

Merke: Falls irgendwie möglich, sollte das Blaupausenkonzept (siehe Abschnitt 3.3) Verwendung finden!

```
<?xml version="1.0" encoding="UTF-8"?>
<PersonenName
  xmlns="http://reference.e-government.gv.at/namespace/persondata/de/20040201#"
  xmlns:dsig="http://www.w3.org/2000/09/xmldsig#"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://reference.e-government.gv.at/namespace/persondata/de/2004-02-01/PersonData.xsd">

  <Vorname>Alexander</Vorname>
  <Familiename>Leiningen-Westerburg</Familiename>
  <Affix typ="akademischerGrad" position="prefix">Mag.</Affix>
  <Affix typ="akademischerGrad" position="prefix">MAS</Affix>
</PersonenName>
```

Beispiel 36: XML Beispieldatei für die Verwendung von Atomaren Elementen

3.6 Alternative Telekom Elemente

365

Englische Version: Telephone ; **Deutsche Version:** Telefon

Englische Version: Mobile ; **Deutsche Version:** Handy

Englische Version: Fax ; **Deutsche Version:** Fax

Englische Version: Pager ; **Deutsche Version:** Pager

Englische Version: TTYTDD ; **Deutsche Version:** TTYTDD

370

Beschreibung: Container für diverse Telekom Elemente

In Anlehnung an den Atomaren Elemente Ansatz können auch alternative Elemente verwendet werden, die nicht direkt in die PersonData Struktur eingebunden sind, im Schema aber definiert werden. Die Schemata aller Alternativen Telekom Elemente sind untereinander ident, weswegen hier nur eines stellvertretend für alle anderen betrachtet werden soll. Diese Struktur entspricht dem des Nummer Elements (siehe Abschnitt 2.1.10).

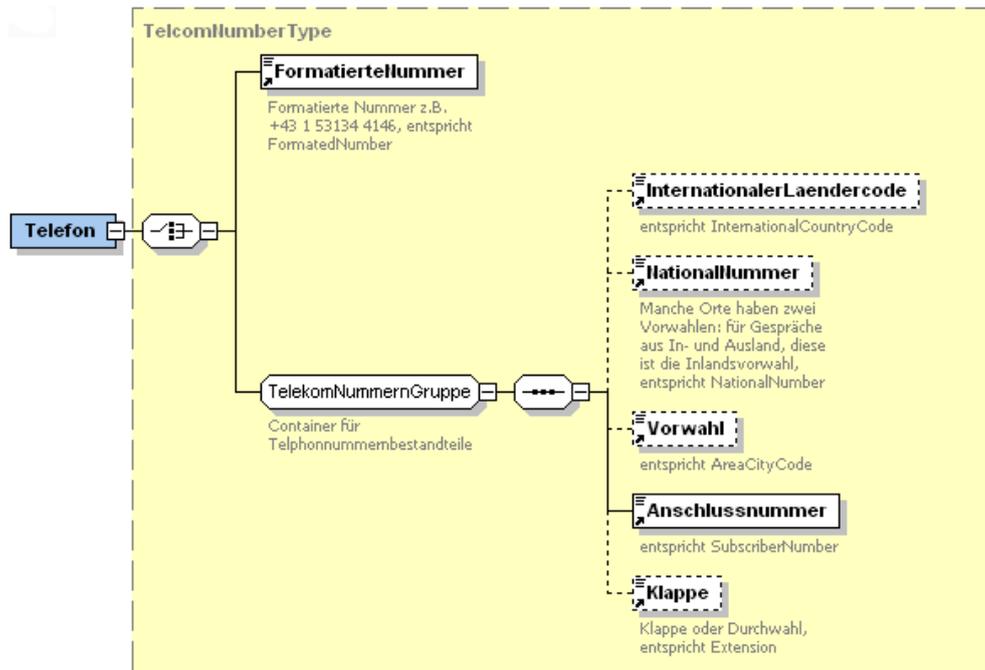


Abbildung 3.10: Unterlemente des Telefon Containers

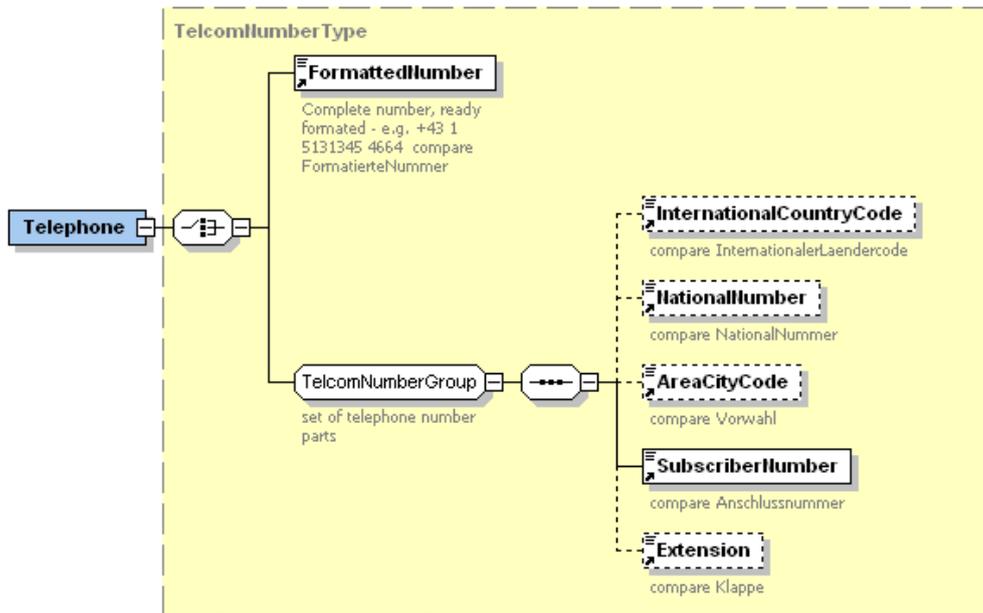


Abbildung 3.11: Unterlemente des Telephone Containers

Englisch	Deutsch	Erklärung
FormattedNumber	FormatierteNummer	Formatierte Nummer, z.B. +43 1 53134 4146
InternationalCoutnryCode	InternationalerLaendercode	Internationaler Ländercode
NationalNumber	Nationalnummer	Manche Orte haben zwei Vorwahlen: für Gespräche aus In- und Ausland, diese ist die Inlandsvorwahl
AreaCityCode	Vorwahl	Vorwahl
SubscriberNumber	Anschlussnummer	Anschlussnummer
Extension	Klappe	Die Klappe (Durchwahl) der Nummer.

Tabelle 3.2: Unterelemente aller alternativen Telekom Elemente

3.7 Welcher Ansatz für welches Problem?

3.7.1 Text- oder Repräsentationslastige Anwendungen

Wenn es sich bei der Anwendung um ein System handelt, das hauptsächlich der Textverarbeitung oder Anzeige von Texten dient, ist es durchaus möglich die gesamte **PersonData** Struktur einzusetzen. Das selbe gilt wenn es sich um Austausch von Informationen in “menschenslesbarer” Form handelt. 380

3.7.2 Webservices

Webservices erfordern i.a. einen hohen Grad an Anpassung an das jeweilige Service. Dieser Anpassungsgrad ist meist nur durch den Blaupausenansatz geboten. Es ist nicht anratenwert als Basis für den Webservice eine gesamte **PersonData** Struktur zu wählen (es sei denn man benötigt wirklich alle Elemente). Keinesfalls sollten Einschränkungen der **PersonData** Struktur an dritter Stelle (z.B. in der Spezifikation des Webservices) hinterlegt werden. 385

Ist ein Einsatz einer Blaupausenlösung aus technischer Sicht zu aufwendig, können in Ausnahmefällen auch “Atomare Elemente” zum Einsatz kommen (siehe Abschnitt 3.5). 390