Postal Address 1.2 (Single Namespace Edition)

Recommendation, 2003 February 26

This version:
PostalAddress.doc

Previous version:
PostalAddress-1_2.doc

Editor:
Kim Bartkus
Paul Kiel
Mark Marsden

Authors:
Members of the Cross-Process Object work group

Contributors:
Members of the Cross-Process Object work group

Copyright statement

Abstract

This document provides all necessary documentation for PostalAddress, including schema, definitions, and examples.

Status of this Document

2003-Aug-12: This specification remains unchanged from the 2003-Feb-26 release. The version number and “Single Namespace Edition” have been added to the title page of the documentation in order to delineate it from previous releases where the Consortium used multiple namespaces. In addition, the “version” attribute of the “xsd:schema” element now reflects this same version number.

2003-Feb-26: This Recommendation was previously published as PostalAddress-1_2, 2001-Oct-16. While the targetNamespace and default namespace have been changed, the Recommendation schema has otherwise remained unaltered.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

Table of Contents

1 Overview

1.1 *Objective*

Create a schema design for a postal address that is flexible enough to be a global standard and which may be used within other HR-XML Consortium schemas.

1.2 *Relationship to Business Processes*

The Postal Address schema attempts to create a generalized container that will allow business processes to pass address information reliably and completely, and in a format that can be efficiently processed.

To this end, the container is to be designed to clearly house the various sections that make up a postal address as it is used from country to country, while allowing the country code to indicate to the business process how the address is to be formatted, according to local postal rules.

1.3 *Scope*

1.3.1 Within Scope

- The project will define the Postal Address, which may be used to globally send mail to individuals or organizations.
- The deliverable will be a schema, which may be transformed by a system to a format required for mailing.
- Internal routing will not be defined as separate elements. Information such as mail stop will be included as part of Recipient.
- Some countries can validate an address number within a street if they are separate elements. Version 1.1 will provide for this feature.

1.3.2 Outside of Scope

- This project will not define a location or geo code. These codes typically define latitude and longitude locations.
- Effective dating will not be addressed within this proposal. When effective dating is resolved, this proposal will be re-evaluated to assess the impact.
- This proposal does not recommend nor imply how an address should be stored in a database. It also does not address sorting or reporting formats for an address.

1.4 *Design Requirements*

- Syntax must be self-documenting.
- Must have enough information to be used for global mail delivery.

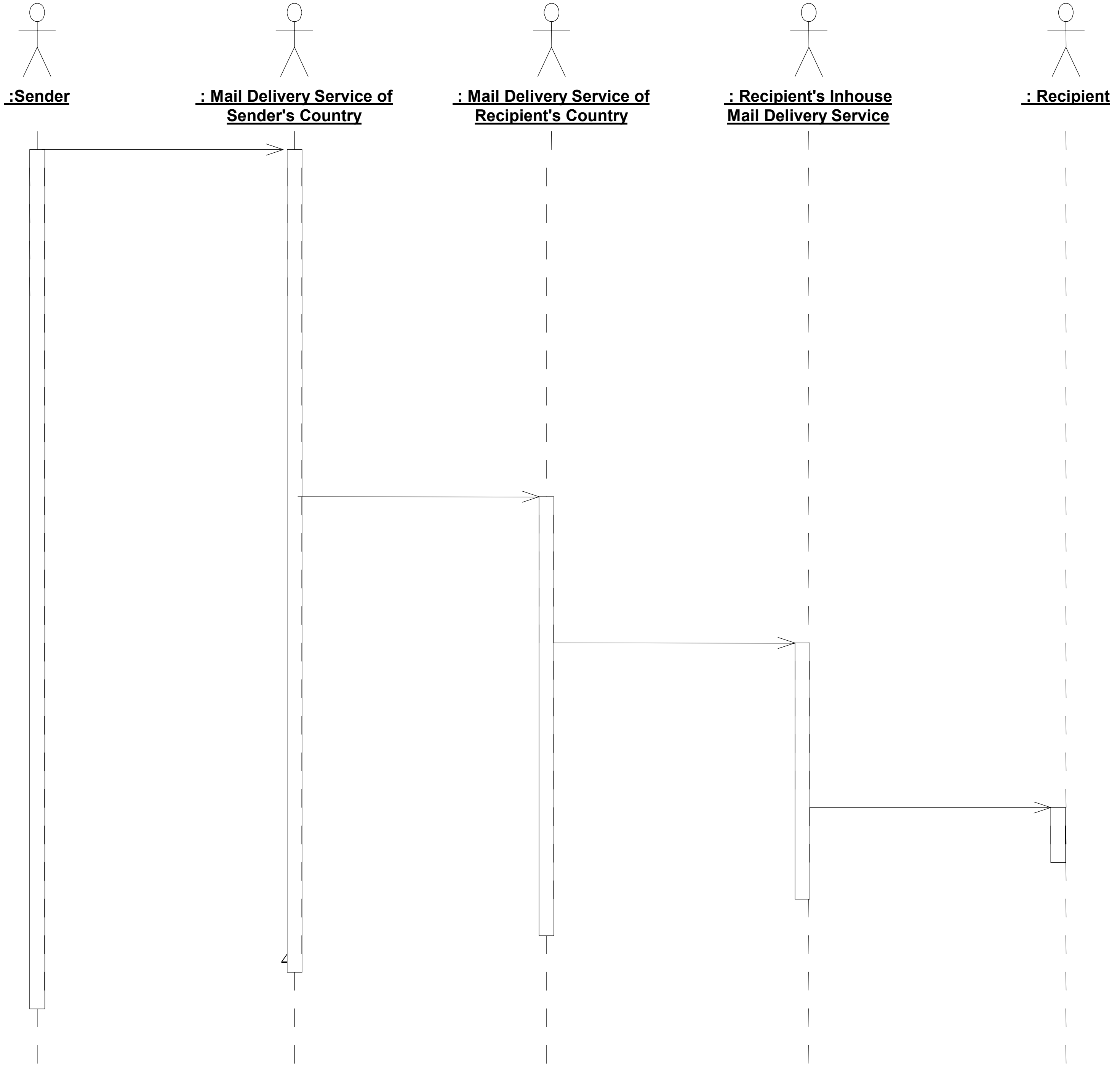
1.5 *Mail Delivery Process*

The sender formats the address according to the rules of the sender's and the recipient's country.
Local rules are used to represent the recipient's country (e.g. recipient's country printed in local language or transformed to a postal code prefix).
The rules of the recipient's country are used to format the rest of the address (e.g. order of city and postal code). For this part, the recipient's language (if given) should be used for city names, street names...

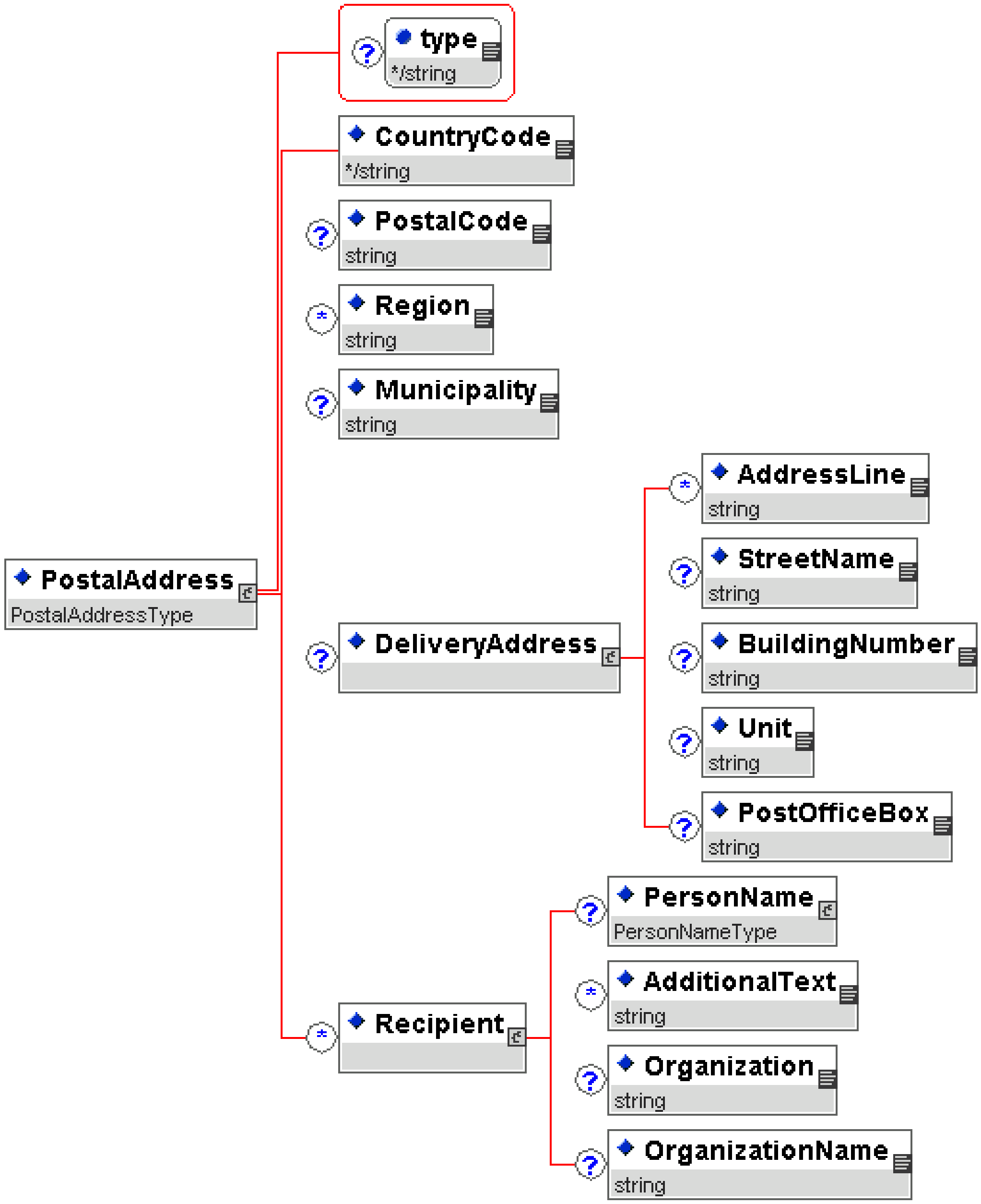
The mail delivery service of the sender's country uses the recipient's country to determine the delivery service of the recipient's country. It delivers the mail to that service.

The mail delivery service of the recipient's country uses postal code, region, municipality, and delivery address to deliver the mail to the inhouse mail delivery service at the recipient's address.

The inhouse mail delivery service uses the recipient information to deliver the mail to the recipient.


:Sender
: Mail Delivery Service of Sender's Country
: Mail Delivery Service of Recipient's Country
: Recipient's Inhouse Mail Delivery Service
: Recipient

2 Schema Design



2.1 *Schema Elements Explained*

Component Name	ContentModel Data type Occurrence: Sequence | Choice | All (minOccurs/maxOccurs) Attributes	Definition
/ **PostalAddress**	- PostalAddressType - (1/1)	Describes a postal address used for delivery of mail.

/ PostalAddress/ **CountryCode**	xsd:restriction **base:** xsd:string	Contains the ISO 3166-1 two-character country code.
/ PostalAddress/ **PostalCode**	- xsd:string - S (0/1)	Codes established by postal authorities for purposes of sorting and delivering mail.
/ PostalAddress/ **Region**	- xsd:string - S (0/*)	Represents the State, Province, and/or County. Military addresses should be stored in multiple regions (hierarchy region from highest to most specific): 1 st region = APO (Army/Airforce Post Office); FPO (Fleet Post Office). 2 nd region = 2 letter designator for part of world (AE Œ Europe, AA Œ Americas)
/ PostalAddress/ **Municipality**	- xsd:string - S (0/1)	Represents the city, town, village, or hamlet.
/ PostalAddress/ **DeliveryAddress**	AddressLine - xsd:string - S (0/*) StreetName - xsd:string - S (0/1) BuildingNumber - xsd:string - S (0/1) Unit - xsd:string - S (0/1) PostOfficeBox - xsd:string - S (0/1)	Contains one formatted address line with all of its pieces in their proper place. This includes all of the necessary punctuation. This de-normalized form of the delivery address cannot be easily parsed. AddressLine is used for delivery by the postal service. May contain the name or number of the building, house, and/or street. If the address is decomposed into StreetName and BuildingNumber, do not use AddressLine to store the address. Examples may include: Hancock Building; 5223 Oak Street; 213; East 23rd Avenue; P.O. Box 241; Suite 200.
/ PostalAddress/ DeliveryAddress/ **AddressLine**	- xsd:string - S (0/*)	AddressLine is used for delivery by the postal service. May contain the name or number of the building, house, and/or street. If the address is decomposed into StreetName and BuildingNumber, do not use AddressLine to store the address. Examples may include: Hancock Building; 5223 Oak Street; 213; East 23rd Avenue; P.O. Box 241; Suite 200.
/ PostalAddress/ DeliveryAddress/ **StreetName**	- xsd:string - S (0/1)	Contains the street name or number. This may be used for verification, building the address, or storing in a database. If the address is decomposed into StreetName, BuildingNumber and Unit, do not use AddressLine to duplicate that part of the address information. Examples may include: Oak Street; East 23rd Avenue.
/ PostalAddress/ DeliveryAddress/ **BuildingNumber**	- xsd:string - S (0/1)	This element is defined as a string to handle "numbers" such as 7A or 15/III. The term "BuildingNumber" was also used instead of HouseNumber so all types of buildings could apply (house, building, warehouse, tower, etc). This may be used for verification, building the address, or storing in a database. If the address is decomposed into StreetName, BuildingNumber and Unit, do not use AddressLine to duplicate that part of the address information. Examples may include: Hancock Building; 5223.
/ PostalAddress/ DeliveryAddress/ **Unit**	- xsd:string - S (0/1)	Contains the Apartment, Suite, Unit, Room, Floor, Trailer, Level, Hanger, etc. This may be used for verification, building the address, or storing in a database. If the address is decomposed into StreetName, BuildingNumber and Unit, do not use AddressLine to duplicate that part of the address information. Examples may include: Apt. 124, Ste. 300, Upper, Hanger A.
/ PostalAddress/ DeliveryAddress/	- xsd:string - S (0/1)	Contains the Post Office Box. This may be used for verification, building the address, or storing in a

PostOfficeBox		database. Example: P.O. Box 241.
/ PostalAddress/ **Recipient**	PersonName - PersonNameType - S (0/1) AdditionalText - xsd:string - S (0/*) Organization - xsd:string - S (0/1) OrganizationName - xsd:string - S (0/1)	Contains information about the recipient. This may include a person's name, an organization name, and/or additional information.
/ PostalAddress/ Recipient/ **PersonName**	- PersonNameType - S (0/1)	The name of a person.
/ PostalAddress/ Recipient/ **AdditionalText**	- xsd:string - S (0/*)	May contain other recipient routing information in addition to organization and person name. AdditionalText is used for further routing after it has been delivered by the postal service.
/ PostalAddress/ Recipient/ **Organization**	- xsd:string - S (0/1)	Information identifying the organization for which the enrollment data is being transmitted.
/ PostalAddress/ Recipient/ **OrganizationName**	- xsd:string - S (0/1)	Contains information about the recipient. This may include a person's name, an organization name, and/or additional information.

/ [PostalAddressType] / **type**	xsd:restriction **base:** xsd:string [Enumerations]: postOfficeBoxAddress, streetAddress, militaryAddress, undefined	Defines if the postal address is a street address, military, or post office box.
		type = postOfficeBoxAddress
		type = streetAddress
		type = militaryAddress
		type = undefined (default)

3 Reference Examples

Contains examples for sending mail within the same country or sending from country to country. When sending mail from one country to another, the country must be written in the sender's language. All other parts of the postal address may be written in the receiver's language. When sending mail within the same country, the country name may or may not be used.

United States	Mailstop: B1-210
	Karen Barber
	Market Surveyors
	2455 University Blvd
	Denver, CO 80237
	USA
	
	<**PostalAddress**>
	  <**CountryCode**>US**</CountryCode>**
	  <**PostalCode**>80237 **</PostalCode>**
	  <Region>CO**</Region>**
	  <**Municipality**>Denver**</Municipality>**
	 <**DeliveryAddress**>
	    <**AddressLine>**2455 University Blvd</**AddressLine>**
	  </DeliveryAddress>
	  <**Recipient**>
	    <**PersonName**>
	      <FormattedName>Karen Barber**</FormattedName>**
	    </PersonName>
	    <AdditionalText>Mailstop: B1-210**</AdditionalText>**
	    <OrganizationName>Market Surveyors**</OrganizationName>**

```
  </Recipient>
</PostalAddress>
```

Santhi Mwanza
MinnBest Corp.
4982 E Beauregard Ave.
Minneapolis, MN
50493-1234

```
<PostalAddress>
  <CountryCode>US</CountryCode>
  <PostalCode>50493-1234</PostalCode>
  <Region>MN</Region>
  <Municipality>Minneapolis</Municipality>
  <DeliveryAddress>
    <StreetName>E Beauregard Ave.</StreetName>
    <BuildingNumber>4982</BuildingNumber>
  </DeliveryAddress>
  <Recipient>
    <PersonName>
      <GivenName>Santhi</GivenName>
      <FamilyName>Mwanza</FamilyName>
    </PersonName>
    <OrganizationName>MinnBest Corp.</OrganizationName>
  </Recipient>
</PostalAddress>
```

Sarah Olson
c/o Mrs. Cole
2000 Merrill Lane
Sunnyvale, CA
93121

```
<PostalAddress>
  <CountryCode>US</CountryCode>
  <PostalCode>93121</PostalCode>
  <Region>CA</Region>
  <Municipality>Sunnyvale</Municipality>
  <DeliveryAddress>
    <AddressLine>c/o Mrs. Cole</AddressLine>
    <AddressLine>2000 Merrill Lane</AddressLine>
  </DeliveryAddress>
  <Recipient>
    <PersonName>
      <FormattedName>Sarah Olson</FormattedName>
    </PersonName>
  </Recipient>
</PostalAddress>
```

H.L. Mencken
PO Box 350
Hollywood, CA 93029-1200

```
<PostalAddress type="postOfficeBoxAddress">
  <CountryCode>US</CountryCode>
```

	<**PostalCode**>93029-1200</**PostalCode**> <**Region**>CA</**Region**> <**Municipality**>Hollywood</**Municipality**> <**DeliveryAddress**> <**PostOfficeBox**>PO Box 350</**PostOfficeBox**> </**DeliveryAddress**> <**Recipient**> <**PersonName**> <**FormattedName**>H.L. Mencken</**FormattedName**> </**PersonName**> </**Recipient**> </**PostalAddress**> Major Carmen Li 111th Maint Co Unit 342 APO AA 00932 <**PostalAddress** type="militaryAddress"> <**CountryCode**>US</**CountryCode**> <**PostalCode**>00932</**PostalCode**> <**Region**>APO</**Region**> <**Region**>AA</**Region**> **<DeliveryAddress>** <**AddressLine**>111th Maint Co</**AddressLine**> <**AddressLine**>Unit 342</**AddressLine**> </**DeliveryAddress**> <**Recipient**> <**PersonName**> <**GivenName**>Carmen</**GivenName**> <**FamilyName**>Li</**FamilyName**> <**Affix** type="formOfAddress">Major</**Affix**> </**PersonName**> </**Recipient**> </**PostalAddress**>
Germany	Ringstr. 25 D-80395 Frankfurt am Main **<PostalAddress type=”streetAddress”**> **<CountryCode**>DE**</CountryCode>** <**PostalCode**>80395 **</PostalCode>** <**Municipality**>Frankfurt am Main**</Municipality>** <**DeliveryAddress**> <**StreetName>**Ringstr. **</StreetName>** <**BuildingNumber>** 25**</BuildingNumber>** **</DeliveryAddress>** </**PostalAddress**> *Note. The ‘D-’ is a European country code prefix that is added in front of the postal code and is not part of the postal code.*
Finland	**From outside of Finland:** Hallituskatu 6 B 27 FIN-33200 TAMPERE FINLAND

	From within Finland:
	Hallituskatu 6 B 27
	33200 TAMPERE
	
	<**PostalAddress**>
	  <**CountryCode**>FI**</CountryCode>**
	  <**PostalCode**>33200**</PostalCode>**
	  <**Municipality**>TAMPERE **</Municipality>**
	  <**DeliveryAddress**>
	    <**AddressLine>**Hallituskatu 6 B 27<**/AddressLine>**
	  </DeliveryAddress>
	</**PostalAddress**>

Japan	456-0042
	Tokyo-to Shibuya-ku Shibuya
	4-5-5
	
	<**PostalAddress**>
	  <**CountryCode**>JP**</CountryCode>**
	  <**PostalCode**> 456-0042**</PostalCode>**
	  <**Region**>Tokyo-to**</Region>**
	  <Region>Shibuya-ku**</Region>**
	  <Region>Shibuya**</Region>**
	  <**DeliveryAddress**>
	    <**AddressLine>**4-5-5<**/AddressLine>**
	  </DeliveryAddress>
	</**PostalAddress**>

France	Paul Dupont
	66, rue de l'église
	750026 Paris
	(France)
	
	<**PostalAddress**>
	  <**CountryCode**>FR**</CountryCode>**
	  <**PostalCode**>750026**</PostalCode>**
	  <**Municipality**>Paris **</Municipality>**
	  <**DeliveryAddress**>
	    <**AddressLine>**66, rue de l'église**</AddressLine>**
	  </DeliveryAddress>
	  <**Recipient**>
	    <**PersonName**>
	      <**FormattedName**>Paul Dupont**</FormattedName>**
	    </PersonName>
	  </Recipient>
	</**PostalAddress**>

Switzerland	Georges Dupuis Route des Clos 7 2012 Auvernier (Suisse) <**PostalAddress**> <**CountryCode**>CH**</CountryCode>** <**PostalCode>**2012**</PostalCode>** <**Municipality**>Auvernier **</Municipality>** <**DeliveryAddress>**

	<AddressLine>Route des Clos 7 **</AddressLine>** **</DeliveryAddress>** <**Recipient**> <**PersonName**> <**FormattedName**>Georges Dupuis**</FormattedName>** **</PersonName>** **</Recipient>** </**PostalAddress**> *If the letter is mailed from French speaking countries, add Suisse on the last line. If sending from English speaking countries, add Switzerland on the last line.*
Belgium	Albert Dupontel Avenue Lebon 112 boîte 7 1160 Bruxelles (Belgique) <**PostalAddress**> <**CountryCode**>BE**</CountryCode>** <**PostalCode**>1160**</PostalCode>** <**Municipality>**Bruxelles**</Municipality>** <**DeliveryAddress>** **<AddressLine**>Avenue Lebon 112 boîte 7 **</AddressLine>** **</DeliveryAddress>** <**Recipient**> <**PersonName**> <**FormattedName**>Albert Dupontel**</FormattedName>** **</PersonName>** **</Recipient>** </**PostalAddress**> *If the letter is mailed from French-speaking countries, add Belgique on the last line. If sending from English speaking countries, add Belgium on the last line*.
China	**In Roman alphabet:** Mr. Wang, Tai-sheng 4th Floor, #6, lane 15, Alley 283, Section 1, Hsin Sheng South Rd., Taipei, ROC 12345 <**PostalAddress**> <**CountryCode**>CN**</CountryCode>** <**PostalCode**>12345**</PostalCode>** <**Municipality>**Taipei**</Municipality>** <**DeliveryAddress>** **<AddressLine**>4th Floor, #6, lane 15, Alley 283,**</AddressLine>** <**AddressLine**>Section 1, Hsin Sheng South Rd.,**</AddressLine>** **</DeliveryAddress>** <**Recipient**> <**PersonName**> <**FormattedName**>Mr. Wang, Tai-sheng**</FormattedName**> </**PersonName**> **</Recipient>** </**PostalAddress**>

South Korea	**Outside of South Korea:**
	
	345 Bomun-Dong Sungbuk-Gu Seoul, Korea 136-086
	
	<**PostalAddress**>
	  <**CountryCode**>KR**</CountryCode>**
	  <**PostalCode**>136-086**</PostalCode>**
	  <**Region**>Seoul**</Region>**
	  <**Region**>Sungbuk-Gu **</Region>**
	  <**DeliveryAddress>**
	    <AddressLine>345 Bomun-Dong **</AddressLine>**
	  </DeliveryAddress>
	</**PostalAddress**>

Mexico	Hacienda de Corralejo Nº 5
	Bosques de Echegaray
	13355 Naucalpan, Edo de México
	
	<**PostalAddress**>
	  <**CountryCode**>MX **</CountryCode>**
	  <**PostalCode**>13355**</PostalCode>**
	  <**Municipality**>Naucalpan**</Municipality>**
	  <**DeliveryAddress>**
	    <AddressLine>Hacienda de Corralejo Nº 5 **</AddressLine>**
	    <**AddressLine**>Bosques de Echegaray **</AddressLine>**
	  </DeliveryAddress>
	</PostalAddress>

Spain	28 C/*Alameda 28034 Colmenar, Madrid <**PostalAddress**> <**CountryCode**>ES**</CountryCode>** <**PostalCode**>28034**</PostalCode>** **<Region>**Colmenar**</Region >** <**Municipality**>Madrid**</Municipality>** <**DeliveryAddress>** **<AddressLine**>28 C/*Alameda **</<AddressLine>** **</DeliveryAddress>** **</PostalAddress>** *These abbreviations stand for:* *Avenue = Avda* *Boulevard = Blvr* *Street = C* *Paseo = Pº* *Should be written in front of the name, followed by /.*
Argentina	Las Heras 1045 piso 3 departamento A 1181 Buenos Aires Argentina <**PostalAddress**> <**CountryCode**>AR**</CountryCode>** <**PostalCode**>1181**</PostalCode>**

	<**Municipality**>Buenos Aires**</Municipality>** <**DeliveryAddress>** **<AddressLine**>Las Heras 1045 piso 3 departamento A**</AddressLine>** **</DeliveryAddress>** **</PostalAddress>**
Brazil	Rua Francisco Deslandes, nº 470 apto. 201. Bairro Anchieta. Belo Horizonte, Minas Gerais. Brasil. 30320-500 <**PostalAddress**> <**CountryCode**>BR**</CountryCode>** <**PostalCode**>30320-500**</PostalCode>** <**Region**> Minas Gerais.**</Region>** <**Municipality**>Belo Horizonte**</Municipality>** <**DeliveryAddress>** **<AddressLine**>Rua Francisco Deslandes, nº 470**</AddressLine>** <**AddressLine**>apto. 201. Bairro Anchieta.**</AddressLine>** **</DeliveryAddress>** </**PostalAddress**>
Italy	Via Trento 34 43036 Fidenza (PR) Italy <**PostalAddress**> <**CountryCode**>IT**</CountryCode>** <**PostalCode**>43036**</PostalCode>** <**Region**>PR**</Region>** <**Municipality**>Fidenza**</Municipality>** <**DeliveryAddress>** **<AddressLine**>Via Trento 34**</AddressLine>** **</DeliveryAddress>** **</PostalAddress>** *First line contains the street address (name and number)* *Second line contains the C.A.P (postal code), City, and province (in parenthesis).* *Third line contains country, if mailed outside of country.*
Canada	Paul Mercier 101, rue des Pins, app.10 BEAUPORT, Québec G1E 1K3 (Canada) <**PostalAddress**> <**CountryCode**>CA**</CountryCode>** <**PostalCode**>G1E 1K3**</PostalCode>** <**Region**>Québec**</Region>** <**Municipality**>BEAUPORT **</Municipality>** <**DeliveryAddress>** **<AddressLine**>101, rue des Pins, app.10**</AddressLine>** **</DeliveryAddress>** <**Recipient**>

	<**PersonName**> <**FormattedName**>Paul Mercier**</FormattedName>** **</PersonName>** **</Recipient>** </**PostalAddress**> *Canadian postal codes are always listed in the same format: The sequence is always Alphabetical character/Number/Alpha (full space) Number/Alpha/Number. Each code represents a specific geographic location, ranging from one side of a city block to a specific organization, which receives large volumes of mail.*
Russia	308061, Belgorod А.Я. 495 **<PostalAddress type=”postOfficeBoxAddress”**> **<CountryCod**e>RU**</CountryCode>** **<PostalCod**e>308061**</PostalCode>** **<Municipalit**y>Belgorod**</Municipality>** **<DeliveryAddress>** **<AddressLine**> А.Я. 495**</AddressLine>** **</DeliveryAddress>** **</PostalAddress>** *In the example above, “*А.Я.*” stands for “abonentny yashik” which is Russian for “Post Office Box”. The format is inverted – postal code and municipality are on the first lines, followed by the street address and finally by the recipient. The above example is for a delivery address – in a return address the city comes before the postal code.*
Simplified Chinese (Mainland China)	Typical Chinese envelopes have zip code (six digits) boxes at the left upper corner. You should fill them out with recipient’s zip code. Address can consist of three lines or four lines, depending upon your need. First line contains country, city and organization (or “unit” as most Chinese would call it). Second line consists of street and number. The third line prints the recipient’s name. There should always be one or two space between name and the address. The sender’s address should always be printed at the right lower corner on the envelope. The format is the same as above except that the zip code appears at the end of the address and there is no space needed between the name and address

4 Implementation Considerations

- When generating a mailing label, the country code may be translated into a country name based upon the language of the postal sender.
- **Recipent** is optional because many business processes store the intended recipient name separately from postal address. If your schema handles the recipient information elsewhere, avoid redundancy by not populating **Recipient** in **PostalAddress** elements.
- The order of the elements within **PostalAddress** often matters when transforming elements for presentation purposes. For example, multiple **AddressLine** elements typically will be rendered on separate lines in the order that they where given in the XML.
- If the **PostalAddress** includes the name of a person, that information should always be within the **Recipient/PersonName** element and sub elements, never in the **DeliveryAddress** element or **Recipient/AdditionalText** element. The **Recipient/PersonName/FormattedName** element may be used if the XML sender cannot break out individual recipient name parts. Although not preferable, all other recipient information may be sent in **AddressLine** elements when the XML sender does not have enough information to build recipient elements or chooses not to use the **Recipient/PersonName/FormattedName** element. This allows parties receiving XML to safely determine whether or not an address includes the name of the recipient simply by checking for the presence of a **Recipient/PersonName** element.
- The country names are inferred from the ISO 3166 country code standards. It is up to the implementer to provide the cross-reference between the ISO country code and the country name.
- Some shipping companies will not deliver to post office boxes. The **PostalAddress** type is used to differentiate mail sent to a post office box address versus mail send to a street address. This should assist the sender in selecting the appropriate shipping company for delivery to the different types of addresses.
- If an organization has a street and a post office box address, use multiple postal address elements.
- Formatting characters should be left out of the postal codes.
- Organization has been deprecated and is no longer used. The **OrganizationName** should be used instead.
- **StreetName**, **BuildingNumber, Unit** and **PostOfficeBox** allow countries to validate a post office box or an address number within a street.
    - Systems that recognize the **StreetName**, **BuildingNumber, Unit** and **PostOfficeBox** should use these separate elements accordingly and only use **AddressLine** for additional information. A receiving system may use the separate elements for validation purposes, to build the entire address, or to store in the corresponding fields of their database.
    - Systems that don't recognize the separate elements should use the **AddressLine**.
    - Addresses that are decomposed into **StreetName**, **BuildingNumber** and **Unit** should not also be stored in **AddressLine**. It is redundant and may be confusing to send the address using both methods.
- The **PersonName** v01.01 module has been inserted into the **PostalAddress** module to replace the **PersonName** v1.0.
- 'Care Of' data should be part of the **DeliveryAddress**, not part of **Recipient**. When using 'care of information, it should be held in the **AddressLine** element to avoid problems when building the address.
- If Mailstop is used internally to further define where the recipient is, Mailstop should be held in **AdditionalText** element. If Mailstop is used as part of an external mail company, Mailstop should be held in the **AddressLine**.
- The US Postal Service would parse an address as follows:
    1. AdditonalText 1 to many in order given
    2. PersonName
    3. Organization

4. AddressLine 1 to many in order given
5. BuildingNumber StreetName Unit
6. Municipality, Region PostalCode
7. Country substitution for CountryCode

5 Appendix A – Schema Revision History

Version	Date	Revisions
1_2	2001-08-20	Initial draft based on version 1_1 specification
1_2	2001-09-04	Modified Schema design to change **PostalAddress** type to **PostalAddressType.** Updated all references throughout the document. Also update schema design to include **PersonNameType** (was **PersonName** type)
1_2	2001-Oct-16	Approved Recommendation by HR-XML Consortium
1.2	2003-Jan-31	Changed default/target namespaces.
1.2	2003-Feb-26	Approved recommendation by HR-XML Consortium. The default and targetNamespaces of all HR-XML schemas have been standardized to "http://ns.hr-xml.org". This recommendation is available as part of the HR-XML 2_0 architecture.

6 Appendix B – Mailing label transformation reference table

This table provides implementation considerations on how to print the labels.

- ISO 3166-1 contains a two-letter code (Alpha-2-code), a three-letter code (Alpha-3-code) and a three-digit numeric code (Numeric-3-code) for every entry in its list of country names. The HR-XML standard will utilize the ISO 3166-1993 (E) two-letter alpha code.
- This table is a representative sample of mailing label transformations. For current and complete information, please refer to <http://www.oasis-open.org/cover/country3166.html>.

Country	ISO 3166-1
Argentina	AR
Australia	AU
Belgium	BE
Brazil	BR
Canada	CA
China	CN
Cuba	CU
Finland	FI
France	FR
Germany	DE
Hungary	HU
India	IN
Ireland	IE
Italy	IT
Japan	JP
Korea, South	KR
Mexico	MX
South Africa	ZA
Spain	ES
Switzerland	CH
United Kingdom	GB
United States of America	US

7 Appendix C – Sample XSL template

This appendix contains some sample XSL, which might be used to generate envelope addresses for PostalAddress XML data being sent and received within the U.S.A.

For any given PostalAddress, the XSL sample generates lines for the recipient. It then generates delivery address lines, followed by a line containing city, state, and zip. The output is generated in HTML.

For an XML input like this:

```
<PostalAddress>
    <CountryCode>US</CountryCode>
    <PostalCode>07090</PostalCode>
    <Region>NJ</Region>
    <Municipality>Westfield</Municipality>
    <DeliveryAddress>
        <StreetName>St. Marks Ave</StreetName>
        <BuildingNumber>423</BuildingNumber>
    </DeliveryAddress>
    <Recipient>
        <PersonName>
            <GivenName>John</GivenName>

    <PreferredGivenName>Jack</PreferredGivenName>
            <MiddleName>Smith</MiddleName>
            <FamilyName
primary="undefined">Major</FamilyName>
            <Affix type="formOfAddress">Mr.</Affix>
        </PersonName>
        <AdditionalText>Mailstop X37</AdditionalText>
        <OrganizationName>Laptop Welders,
Inc.</OrganizationName>
    </Recipient>
</PostalAddress>
```

The stylesheet generates:

Mr. Jack Major
Mailstop X37
Laptop Welders, Inc.
423 St. Marks Ave
Westfield, NJ 07090

Here is the stylesheet:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
    <xsl:template match="/">
        <xsl:apply-templates/>
    </xsl:template>
    <xsl:template match="PostalAddress ">
        <xsl:apply-templates select="Recipient"/>
        <xsl:apply-templates select="DeliveryAddress"/>
        <xsl:value-of select="Municipality"/>,  <xsl:value-of select="Region"/>
        <xsl:value-of select="PostalCode"/>
        <br/>
    </xsl:template>
    <xsl:template match="DeliveryAddress">
        <xsl:for-each select="AddressLine">
            <xsl:value-of select="."/>
            <br/>
        </xsl:for-each>
        <xsl:if test="PostOfficeBox">
        PO Box <xsl:value-of select="PostOfficeBox"/>
            <br/>
        </xsl:if>
        <xsl:if test="StreetName">
            <xsl:value-of select="BuildingNumber"/>
            <xsl:value-of select="StreetName"/>
            <br/>
        </xsl:if>
    </xsl:template>
    <xsl:template match="Recipient">
        <xsl:apply-templates/>
    </xsl:template>
    <xsl:template match="PersonName">
        <xsl:apply-templates select="Affix[@type='formOfAddress' or @type='aristocraticTitle']"/>
        <xsl:choose>
            <xsl:when test="PreferredGivenName">
                <xsl:value-of select="PreferredGivenName"/>
            </xsl:when>
            <xsl:otherwise>
                <xsl:apply-templates select="GivenName"/>
                <xsl:apply-templates select="MiddleName"/>
            </xsl:otherwise>
        </xsl:choose>
        <xsl:apply-templates select="Affix[@type='aristocraticPrefix']"/>
        <xsl:apply-templates select="FamilyName"/>
        <xsl:apply-templates select="Affix[@type='generation']"/>
        <xsl:apply-templates select="Affix[@type='qualification']"/>
        <br/>
    </xsl:template>
    <xsl:template match="GivenName">
        <xsl:value-of select="."/>
    </xsl:template>
    <xsl:template match="MiddleName">
        <xsl:value-of select="."/>
    </xsl:template>
    <xsl:template match="FamilyName">
        <xsl:value-of select="@prefix"/>
        <xsl:value-of select="."/>
    </xsl:template>
    <xsl:template match="Affix">
        <xsl:value-of select="."/>
    </xsl:template>
    <xsl:template match="AdditionalText">
        <xsl:value-of select="."/>
        <br/>
    </xsl:template>
    <xsl:template match="OrganizationName">
        <xsl:value-of select="."/>
        <br/>
    </xsl:template>
</xsl:stylesheet>
```

8 Appendix D – References

This section contains links and references for items used within the document.

"Guide to Worldwide Postal-Code & Address Formats". Marian Nelson, Editor.
Nelson Intersearch Company. ISSN: 1072-3862, ISBN: 0-9630677-6-1

US Postal Service Publication 28:
http://pe.usps.gov/cpim/ftp/pubs/Pub28/Pub28.pdf

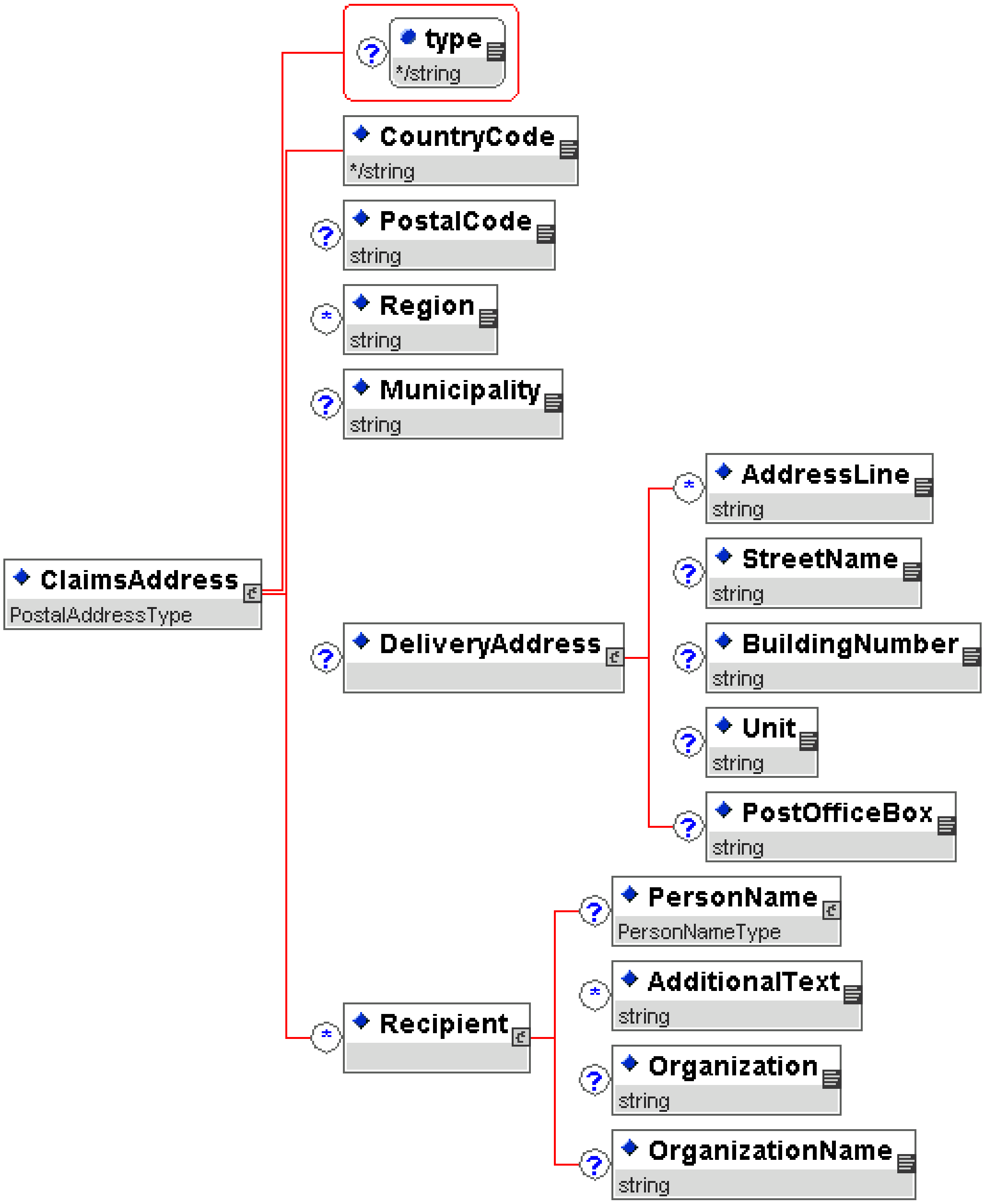
Universal Postal Union links to international postal services:
http://www.upu.int/ap/layout.startup?p_language=AN&p_theme=postadm&p_content_url=/web/an/ServeursAdmin.html

Country Code List: ISO 3166-1993 (E)
<http://www.oasis-open.org/cover/country3166.html>.

9 Appendix E - Schema Examples

9.1 *Example – Using PostalAddressType vs PostalAddress data element*

In the following Schema design, the data element ClaimsAddress has all of the characteristics of PostalAddress, simply by defining ClaimsAddress as type PostalAddressType.

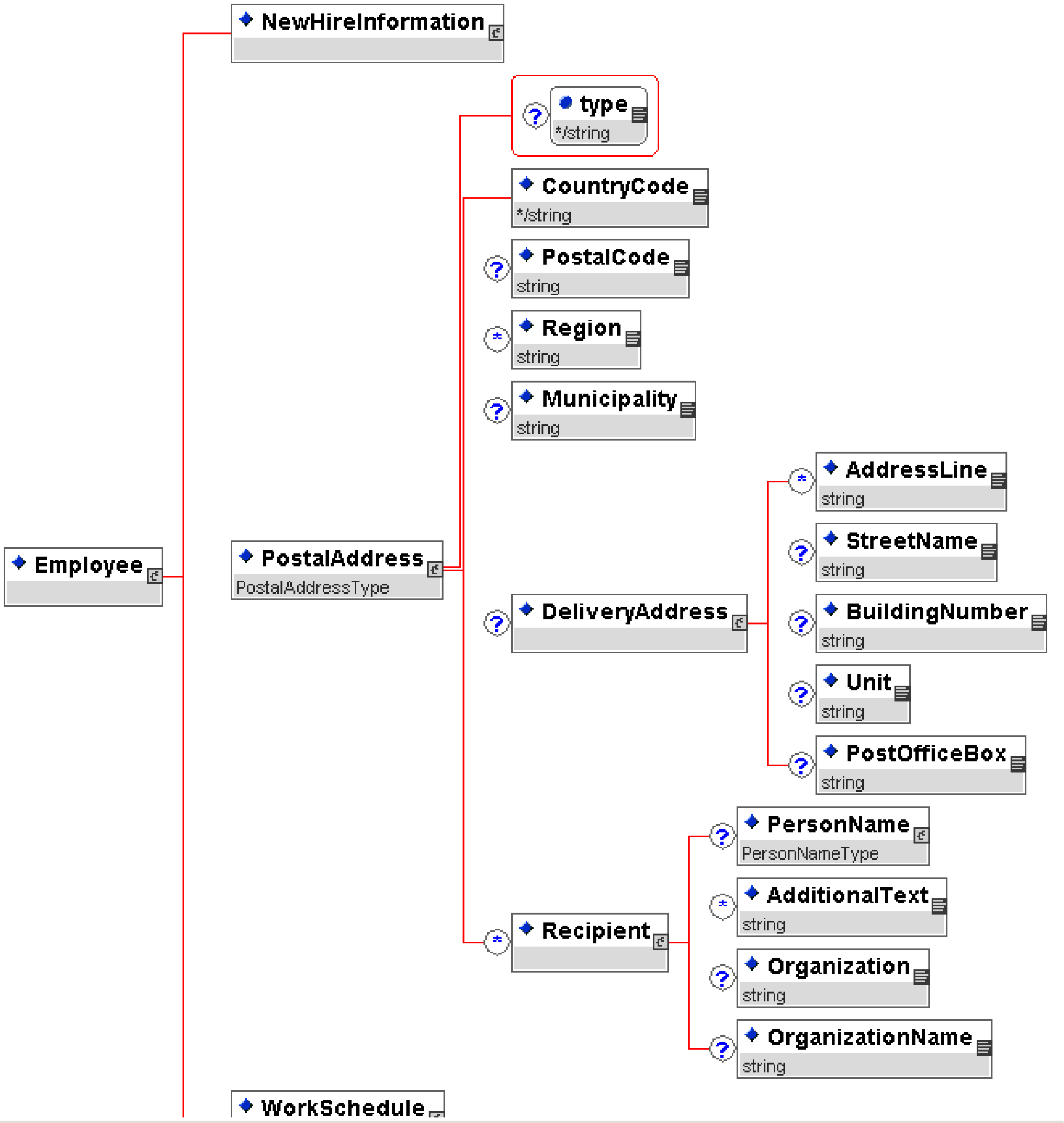


Schema Code:
<xsd:element name = "ClaimsAddress" **type = "PostalAddressType"**/>

Instance Document Example:

```
<ClaimsAddress>
      <CountryCode>US</CountryCode>
      <PostalCode>22153</PostalCode>
      <Region>VA</Region>
      <Municipality>Springfield</Municipality>
      <DeliveryAddress>
         <PostOfficeBox>2345</PostOfficeBox>
      </DeliveryAddress>
</ClaimsAddress>
```

Example 1 continued:
In this example, the PostalAddress data element is just one of many data elements describing the employee.



Schema Code:

```
<xsd:element name = "Employee">
   <xsd:complexType>
     <xsd:sequence>
       <xsd:element name = "NewHireInformation">
          <xsd:complexType>
           <xsd:sequence/>
          </xsd:complexType>
       </xsd:element>
       <xsd:element ref = "PostalAddress"/>
       <xsd:element name = "WorkSchedule">
          <xsd:complexType>
           <xsd:sequence/>
          </xsd:complexType>
       </xsd:element>
     </xsd:sequence>
   </xsd:complexType>
</xsd:element>
```

Instance Document Example:

```
<Employee>
        <NewHireInformation/>
        <PostalAddress>
            <CountryCode>US</CountryCode>
            <PostalCode>22153</PostalCode>
            <Region>VA</Region>
            <Municipality>Springfield</Municipality>
            <DeliveryAddress>
                <PostOfficeBox>1243</PostOfficeBox>
            </DeliveryAddress>
        </PostalAddress>
        <WorkSchedule/>
</Employee>
```