

Cabinet Office Office of the e-Envoy

Towards e-government

UK Online – Information Architecture – Address and Personal Details Fragment

Document Type: Specification

Version: 0.1f

Status: Draft

Document Reference: []

29 March 2001

Document Control

Contributors		
Name/Position	Organisation	Contact Details
Paul Spencer	Boynings/IR	paul.spencer@boynings.co.uk
Chris Gahan	BT	Chris.gahan@bt.com
Ian Jones	BT BS	ian.c.jones@bt.com
Phil Bull	BT BS	phil.bull@bt.com
Steve Towndrow	BT ExaCT	steve.towndrow@bt.com
Tony Fletcher	BT ExaCT	tony.am.fletcher@bt.com
Paul Muschamp	BT Group Technology	paul.muschamp@bt.com
David Hockley	BT Syntegra	dave.hockley@syntegra.bt.co.uk
Kulbir Kaur	BT Syntegra	kulbir.kaur@syntegra.bt.co.uk
Neil Glover	CCTA	neil.glover@ccta.gov.uk
John Ross	CCTA/SECSTAN	ross@secstan.com
Anwar Choudhury	OeE	achoudhury@cit. gsi.gov.uk
Mike Hudson	OeE	MHudson@cit. gsi.gov.uk
Paul Ukpai	OeE	pukpai@cit. gsi.gov.uk
Sebastian Madden	OeE	smadden@cabinet-office.x.gsi.gov.uk
Graham Beaver	Compaq	graham.beaver@compaq.com
Nik Webster	Compaq	nik.webster@compaq.com
Mike Burston	DSS	
Peter Desmond	DSS	p.desmond@new100.dss.gsi.gov.uk
Ken Clarke	ECSoft	Ken.Clark@ecsoft.co.uk
Richard Taylor-Carr	ECSoft	Richard.Taylor-Carr@ecsoft.co.uk
Simon Slocombe	ECSoft	simon.slocombe@ecsoft.co.uk
Rod Mountney	EDS	rod.mountney@eds.com
David Treder	environment-agency	david.treder@environment-agency.gov.uk
Andrew Coote	ESRI(UK)	acoote@esriuk.com
Clare Samways	ESRI(UK)	csamways@esriuk.com
Paul G Butler	IBM	paul_butler@uk.ibm.com
Nig Greenaway	ICL	nig.greenaway@icl.com
Rosi Somerville	IDeA	Rosi.Somerville@IDEA.gov.uk
Steven Brandwood	IDeA	steven.brandwood@idea.gov.uk
Simon Berlin	LB Lambeth	SBerlin@lambeth.gov.uk
Matt Bishop	Microsoft	matbisho@microsoft.com
Linda Williams	Nineveh Consulting	linda.williams@dial.pipex.com

Abstract

The UK Government is establishing the means for citizens and businesses to be able to transact business with the Government electronically. This will use a variety of delivery mechanisms such as the Internet, public 'kiosks', mobile equipment, and interactive television. This series of documents specifies the general information architecture for the overall system. This document specifies that part of this information architecture that is relevant to services and transactions, which involve change of address details for a citizen, and any others that require postal address, contact details and / or some of the common identifiers used by government departments and agencies and local authorities..

Current Version

Date	Version	Status	Editor/ Author	Comment
29 March 2001	0.1f	Draft	A Fletcher, BT	Changes incorporated to address comments made by Mike Burston and Martin Bryan.

Authorisation

Name	
Organisation	
Date	

Change History				
Date	Version	Status	Editor/ Author	Comment
1 August 2000	0.1a	Draft	A Fletcher, BT	First draft
5 September 2000	0.1b	Draft	A Fletcher, BT	Edited according comments received from the first round of public comment.
7 February 2001	0.1c	Draft	A Fletcher, BT	Changed UML to be syntax neutral and added BS 7666 style address.
12 March 2001	0.1d	Draft	A Fletcher, BT	Changes incorporated to the UML, XML plus minor editorial changes as agreed at the Schema group meeting 6 March 01.
19 March 2001	0.1e	Draft	A Fletcher, BT	Changes incorporated to the XML to reflect the revised UML.

Contents

1.	Requirement Definition.....	5
1.1.	Source of Requirement	5
1.2.	Definition Statement	5
1.3.	Structure	5
1.4.	Usage	5
1.5.	Compliance.....	5
1.6.	Conformance	5
2.	UML.....	5
2.1.	Commentary on the UML for address and personal details	5
2.1.1.	The Package diagram.....	5
2.1.2.	Address and Personal Details data structures package.....	6
2.1.3.	Address and Personal Details specific data types	7
2.1.4.	Common data types.....	7
2.2.	UML Class diagrams for Address and Personal Details Fragment of the Information Architecture	8
2.2.1.	The Package diagram.....	8
2.2.2.	Address and Personal Details data structures package.....	9
2.2.3.	Address and personal details specific data types	17
2.2.4.	Common data types.....	18
3.	XML Script	19
3.1.	Commentary on the XML for Address and Personal Details	19
3.2.	XML Schema for Address and Personal Details	19
3.2.1.	Top level for Address and Personal Details	19
3.2.2.	Address Types.....	20
3.2.3.	BS7666 Address Types	22
3.2.4.	Citizen related structures	25
3.2.5.	Simple types	27
4.	Approval.....	29
5.	Abbreviations.....	30
6.	References	30

Figures

Figure 1	Packages used for Address and Personal Details	8
Figure 2	Citizen Details and Address UML structure	9
Figure 3	International Address UML structure	10
Figure 4	UK Address UML structure	11
Figure 5	BS 7666 Address UML structure	12
Figure 6	Citizen Details UML structure	13
Figure 7	Citizen Name UML structure	14
Figure 8	Citizen Registration UML structure	15
Figure 9	Contact Details UML structure	16
Figure 10	Address and personal details specific data types	17
Figure 11	Common data types	18

1. Requirement Definition

1.1. Source of Requirement

1.2. Definition Statement

1.3. Structure

The notational basis for the schemas used for the UK Government's UK Online information architecture is given directly, and by reference, in the introductory document to this series [2]. It recommends the use of UML [3] to represent the data and its relations, and the use of a mapping to the W3C XML Schema Definition Language (XSDL) [4, 5 & 6]. These XML schemas are used in turn for the representation of this information architecture in a syntactical form that can be used in the specification of messages that realise the interchange of data. This document complies with the guidelines and constraints given in [2].

1.4. Usage

This fragment of the overall Information Architecture includes data structures for address, name, email and other contact details, plus some of the identifiers used by government departments such as unique tax reference, National Insurance number and several others. It is, therefore, expected to be widely used in the construction of messages for use within the UK Online system (plus any other systems that adopt the eGovernment Information Architecture). It should be referenced and the data structures imported into messages wherever the data items covered in this fragment are required.

1.5. Compliance

This specification complies with the provisions in the introductory document to this series [2]. Other specifications that form part of the UK Online document set, particularly those specifying messages for the interchange of information, shall comply with this specification. This specification provides specifications for the data items relevant to citizen change of address, and some other types of transaction, in UML and XML Schema format. The UML class diagrams in the UK Online message specification documents shall be copies of the classes and structures specified in this document, with any restrictions applied as appropriate. (Rules for allowable restrictions and constraints are given in [2].) The XML schema equivalents shall be direct copies, or valid imports, of XML schema fragments from this document.

1.6. Conformance

This specification does not contain any provisions that implementations are required to conform to directly. Implementations shall conform to the relevant corresponding message specifications.

2. UML

2.1. Commentary on the UML for address and personal details

The UML that follows consists only of linked class diagrams including data type classes.

2.1.1. The Package diagram

Figure 1 shows the UML packages that are currently specified in this document. The IA-APD-data structures package is the main one and it contains all the data structures specified in this document. It has dependencies on the two data type packages and these dependencies are shown generally in this figure rather than being shown specifically in all the other figures. It is expanded in figures 2 to 9. The IA-APD specific data types package contains all the data types that are for the address and personal details related items, but are unlikely to be needed elsewhere. It is expanded in figure 10. The IA-Common data types package contains all the data types that are likely to be used by other Information Architecture specifications as well. These may be moved to another

document that specifies common data types in due course (possibly the Government Data Standards Catalogue [7]). It is expanded in figure 11.

2.1.2. Address and Personal Details data structures package

Figure 2 shows the root structure for this fragment of the information architecture. The top-level class *CitizenDetailsAndAddress* is composed from *Address* and optionally *CitizenDetails* (figure 6). *Address* is composed of optional *AddressQualifier* and *EffectiveDate*, and a choice of addresses - *ChoiceAddress*, which is composed of the alternatives *InternationalAddress* (Figure 3) and *UKAddress* (Figure 4).

Figure 3 shows *InternationalAddress*, which is composed of 2 to 5 address lines (*IntAddressLine*) plus an optional *Country* line and an optional international postcode (*AddressPostCodeInternational*).

Figure 4 shows the structure for *UKAddress*. This is composed of the alternatives of a *BS7666Address* and a simplified 5 line address (*5LineAddress*) plus an optional unique property reference number (*UPRN*) and an optional choice of sort codes, *ChoiceSortCodes*, which are *MailSort* or *WalkSort*. (Note: a Mailsort code is a code that aims to ensure that items are properly addressed and offers discounts to organisations using it, and a Walksort provides more detail (and higher discounts). Local authorities extensively use both. The Post Office issues files that are used by the sending organisations to manage the processes of correct addressing and sorting of mail items.) The simplified 5 line address (*5LineAddress*) is composed of 2 to 5 lines of address plus an optional postcode.

Figure 5 shows the structure for a full BS 7666 address.

BS7666 is a British standard comprising four parts covering Street Gazetteers, Land and Property Gazetteers, Addresses and Rights of Way. It describes the data to be held in a Gazetteer and the form of this data, thereby facilitating the sharing of land and property information. The NLPG is the aggregation of Local Land and Property Gazetteers (LLPG) which collectively provide an unambiguous identification of land and property and hence provide access to associated information (see also <http://www.nlpg.org.uk/>).

SAON stands for secondary addressable object and is typically a flat number, a room number or post point. PAON stands for primary addressable object and is typically a building number or house name. A Street_Descriptor is simply the name of a Street. An USRN is a unique Street reference number -- these are usually allocated by Street naming authorities before a Street name has been chosen. They may continue to be used as an alternative to, or as well as the Street name even after a Street name has been allocated.

Figure 6 shows *CitizenDetails*. These are composed of one *CitizenName* structures (see figure 7), *CitizenRegistration* information (see figure 8), an optional *PreferredLanguage*, a set of *ContactDetails* (see figure 9), an optional indication of the sex of the Citizen and an optional indication of the birth date of the Citizen. The preferred language is that for contact with the Citizen several can be offered and the sequence should be taken as significant. Currently the only allowed values are English and Welsh, but it is expected that further languages will be added in the future. A birth date provided by a citizen will initially be signalled as 'not verified'. If verification is offered then this can be indicated in the 'verifiedby' field. The values are currently examples based on current DSS practice (note that the applicable values are still under discussion within the DSS and other departments and the values in this document will be updated when they are agreed).

Figure 7 shows *CitizenName* structure. This consists of single surname, plus optional titles, forenames, suffixes and a requested name. The requested name is used, if how the Citizen wants to be addressed differs from their formal names.

Figure 8 shows the *CitizenRegistration* information structure. This consists of a number of identifiers used by local and central government departments and agencies. The ones currently covered include National Insurance number, unique tax reference, driving licence number, national health number, passport number, and electoral roll number. It is anticipated that further identifiers will be added, as the need arises.

Figure 9 shows the *ContactDetails* information structure. This consists of structures for e-mail address, and telephone / fax number. Optionally data items can be used to indicate which is preferred out of several, and whether the phone or fax number is home or office, and whether it is a mobile number.

2.1.3. Address and Personal Details specific data types

Figure 10 shows the data types referred to as data types of attributes in the other classes specified in this document and only likely to be used in this document. Where the data type is an enumeration type the allowed values are shown explicitly.

2.1.4. Common data types

Figure 11 shows the data types referred to as data types of attributes in the other classes specified in this document and likely to be used by other parts of this Information Architecture. Where the data type is an enumeration type the allowed values are shown explicitly.

2.2. UML Class diagrams for Address and Personal Details Fragment of the Information Architecture

2.2.1. The Package diagram

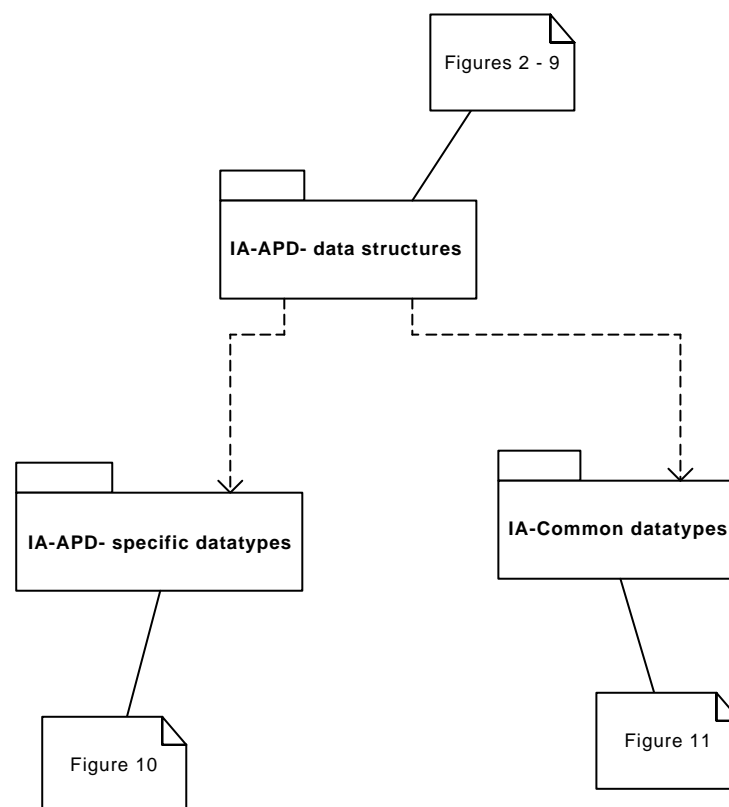


Figure 1 Packages used for Address and Personal Details

2.2.2. Address and Personal Details data structures package

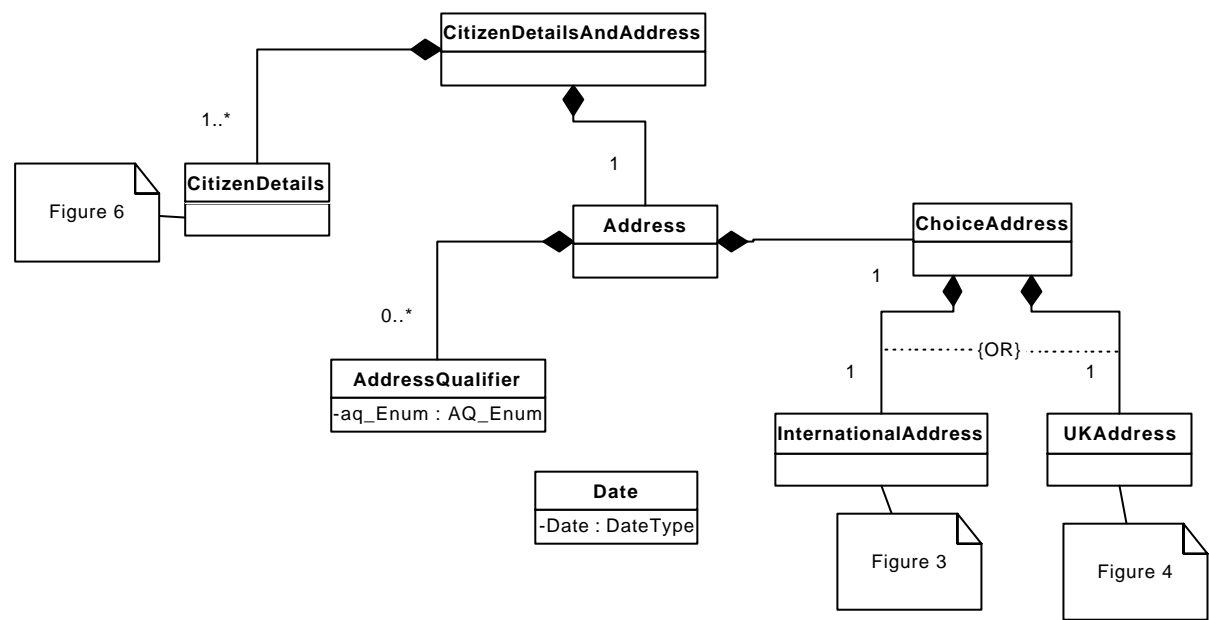


Figure 2 Citizen Details and Address UML structure

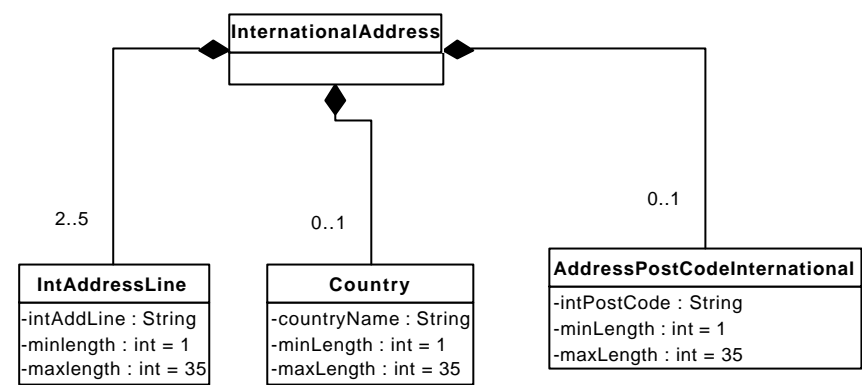


Figure 3 International Address UML structure

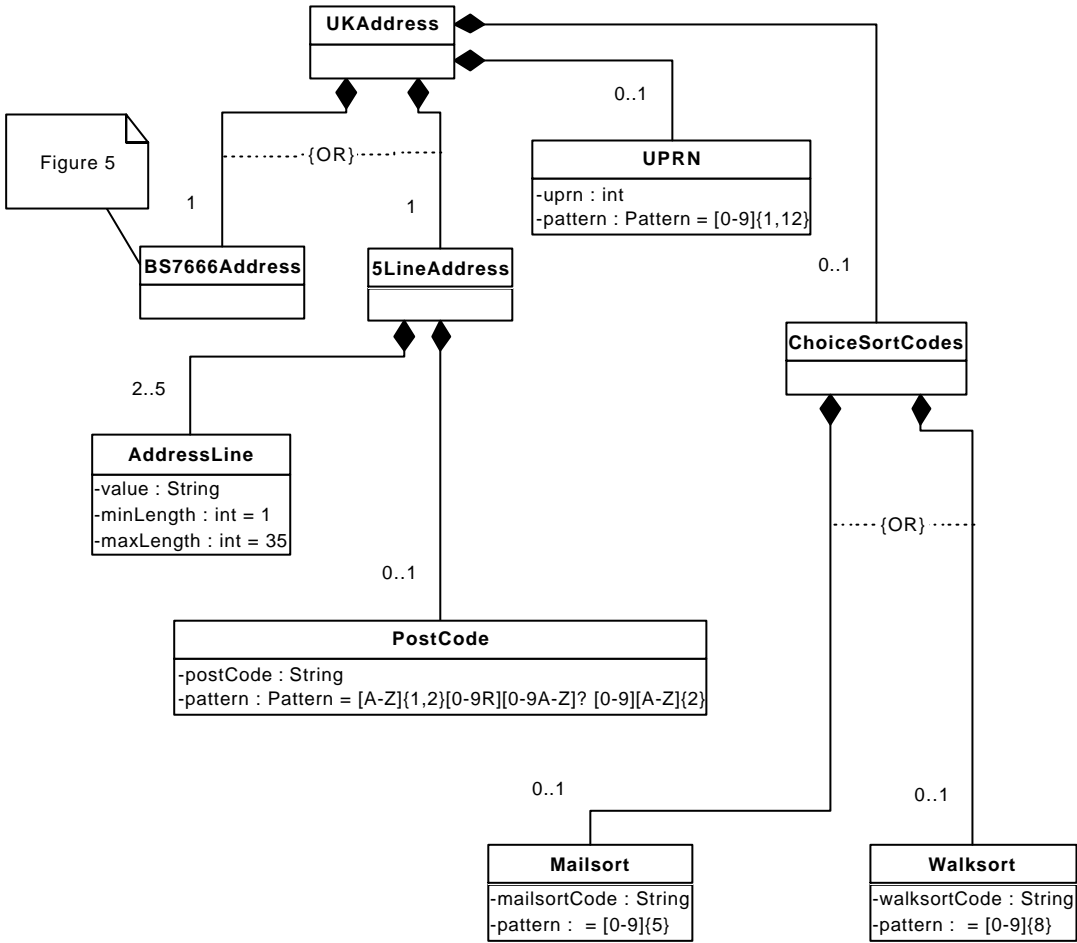


Figure 4 UK Address UML structure

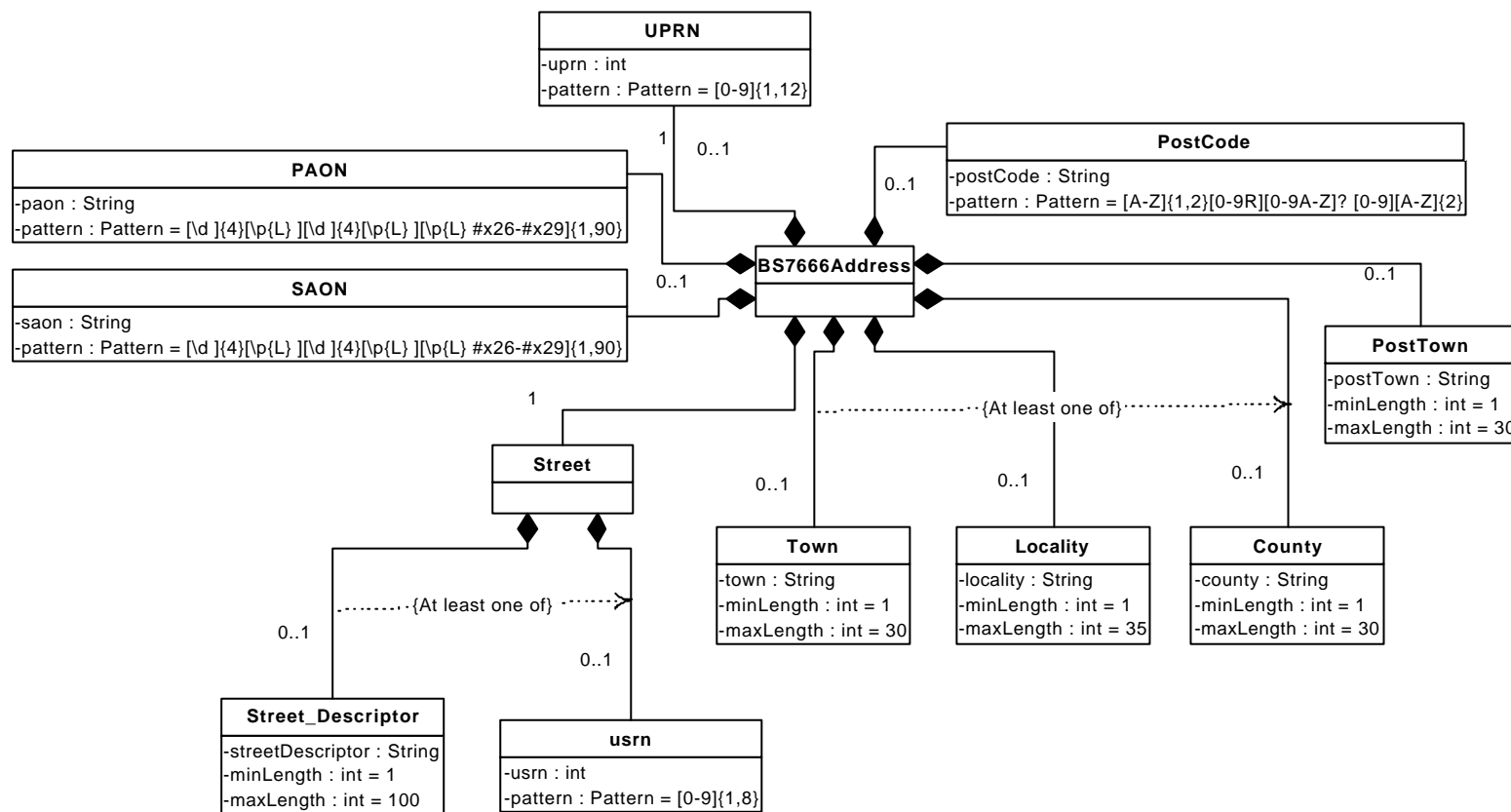


Figure 5 BS 7666 Address UML structure

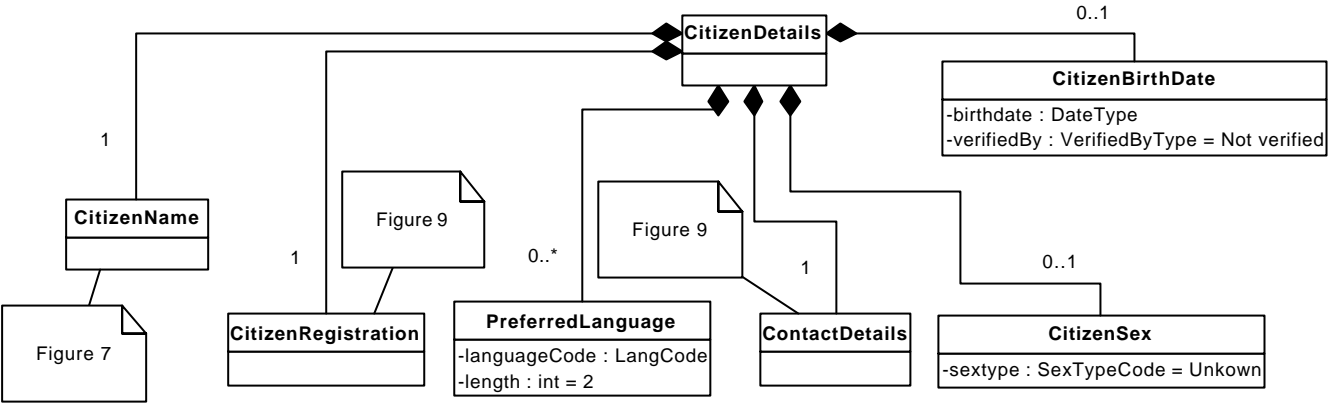


Figure 6 Citizen Details UML structure

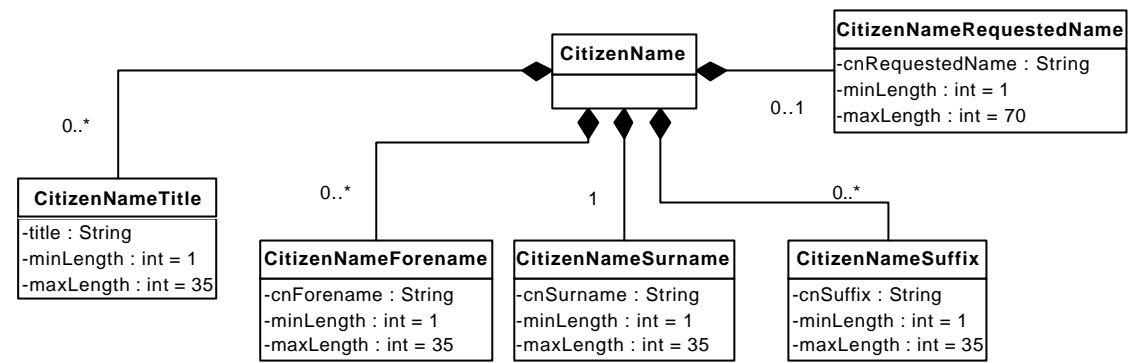


Figure 7 Citizen Name UML structure

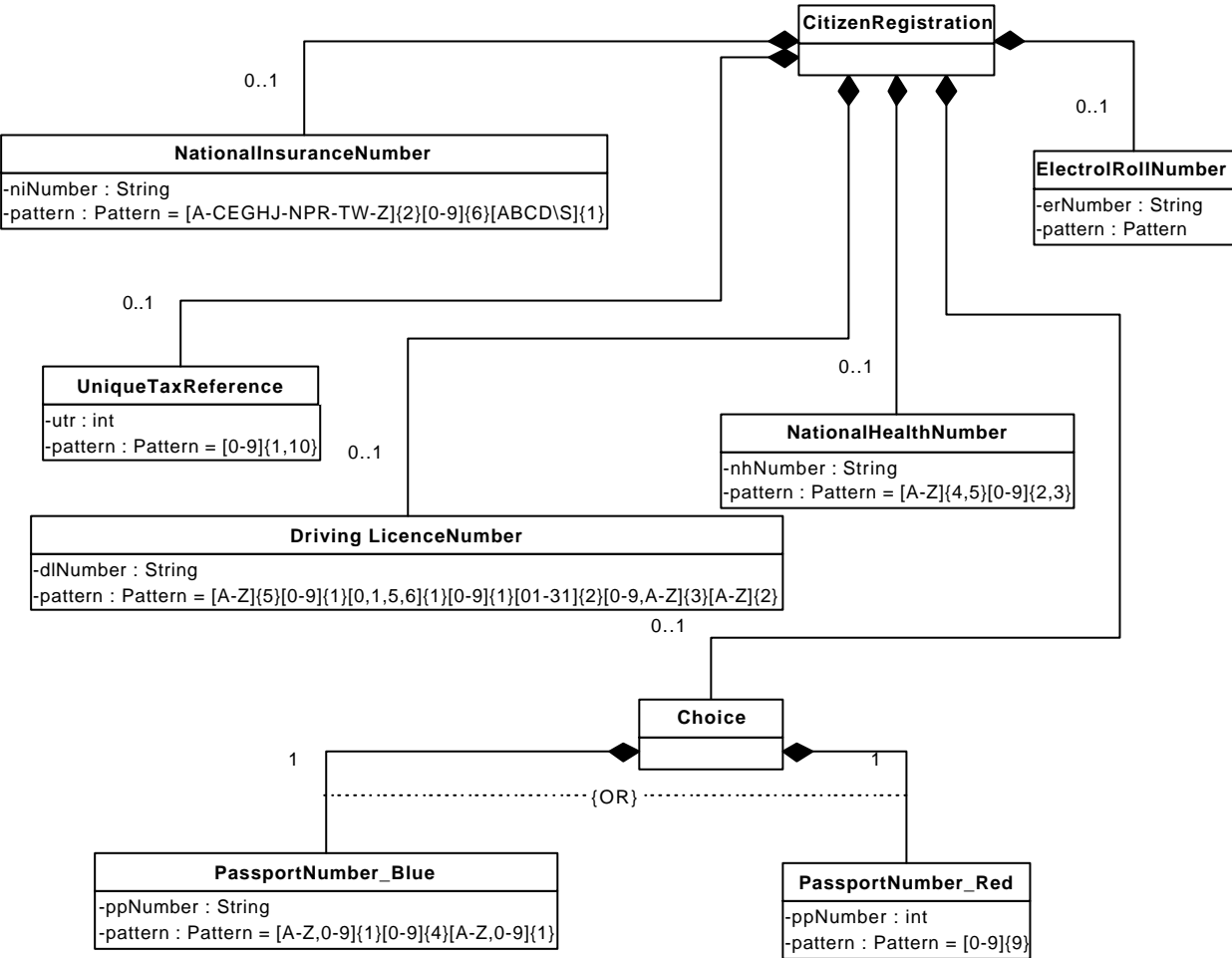


Figure 8 Citizen Registration UML structure

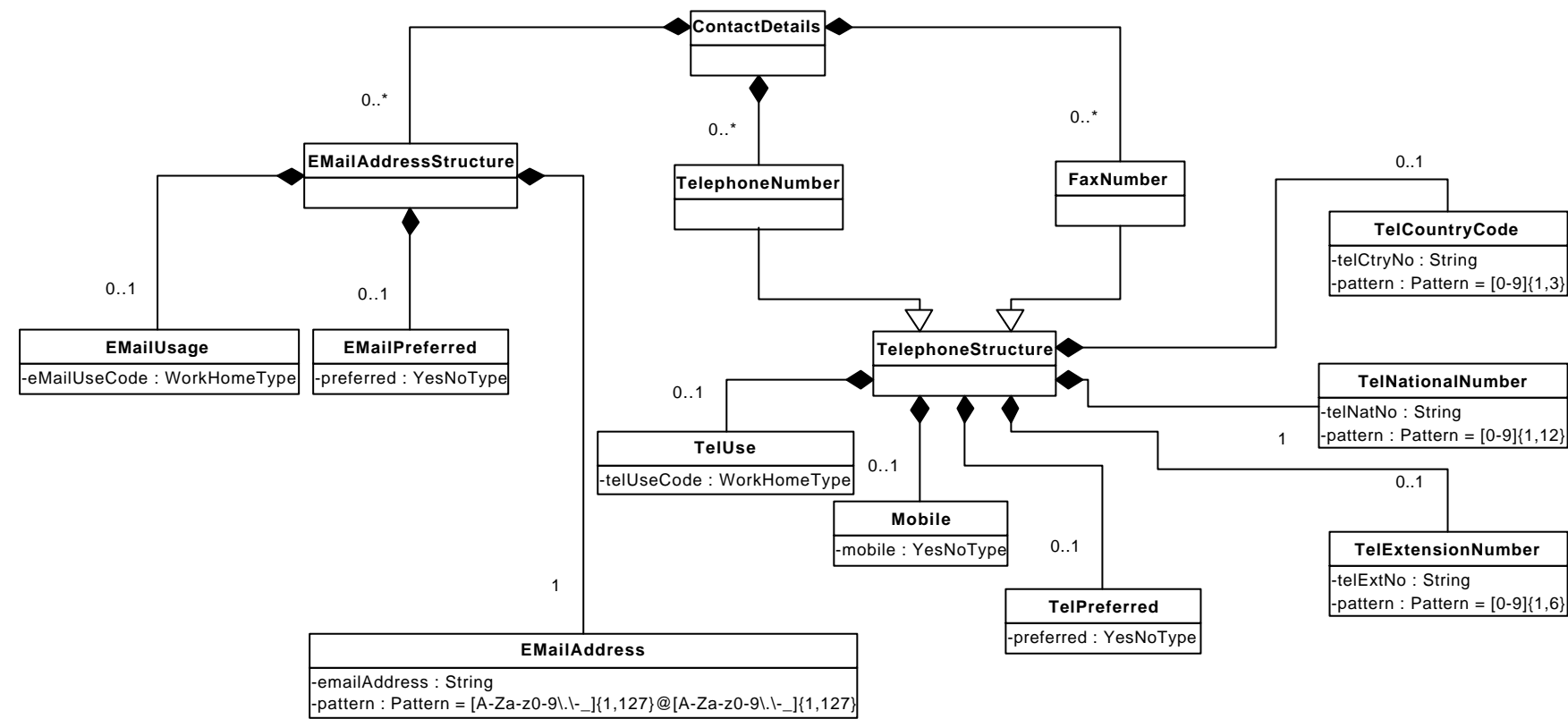


Figure 9 Contact Details UML structure

2.2.3. Address and personal details specific data types

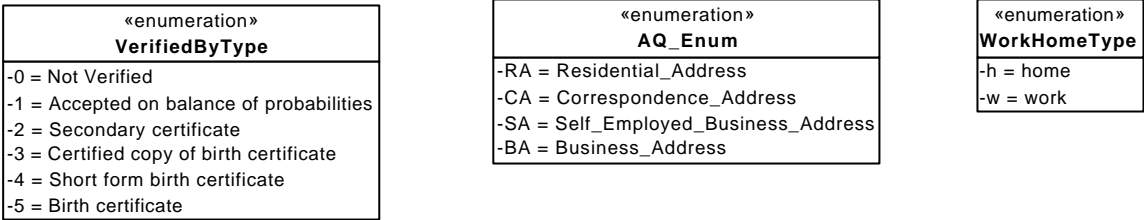


Figure 10 Address and personal details specific data types

2.2.4. Common data types

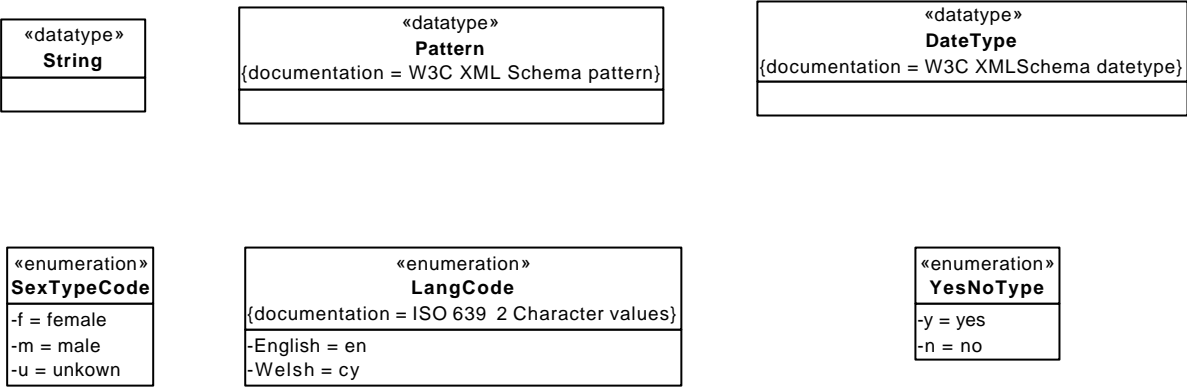


Figure 11 Common data types

3. XML Script

3.1. Commentary on the XML for Address and Personal Details

3.2. XML Schema for Address and Personal Details

3.2.1. Top level for Address and Personal Details

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v3.5 NT (http://www.xmlspy.com) by Adrian Kent
(private) -->
<!-- Schema developed by Paul Spencer, Boynings Consulting
paul.spencer@boynings.co.uk                Amended by Adrian Kent, Office
of the e_Envoy, in accordance with CoA Schema Sub_Group recommendations-->
<xsd:schema targetNamespace="http://GovTalk/Schemas/ChangeOfAddress00d.xsd"
xmlns:xsd="http://www.w3.org/2000/10/XMLSchema"
xmlns:egif="http://GovTalk/Schemas/ChangeOfAddress00d.xsd"
elementFormDefault="qualified" attributeFormDefault="unqualified"
version="1.0e">
  <xsd:include schemaLocation="BS7666Schema3.xsd"/>
  <xsd:include schemaLocation="Citizen.xsd"/>
  <xsd:annotation>
    <xsd:documentation>
      Please note that, in line with the Data Protection Act, it is the
      responsibility of the gateway to deliver to departments only the data
      required for a specific transaction.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:element name="CitizenDetailsAndAddress">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="CitizenDetails"
type="egif:CitizenDetailsStructure" maxOccurs="unbounded">
          <xsd:annotation>
            <xsd:documentation>Should we restrict
to one name for COA?
          </xsd:documentation>
        </xsd:element>
        <xsd:element name="Address"
type="egif:AddressStructure"/>
      </xsd:sequence>
      <xsd:attribute name="EffectiveDate" type="egif:DateType"
use="required"/>
    </xsd:complexType>
  </xsd:element>
  <xsd:complexType name="AddressStructure">
    <xsd:annotation>
      <xsd:documentation>Could possible create an abstract type
GenericAddress which would have to be replaced by a specific address
type</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:annotation>
        <xsd:documentation>BFPO to be added, Effective Date
element can be optionally attached</xsd:documentation>

```

```

        </xsd:annotation>
        <xsd:element name="AddressQualifier" type="egif:AQ_Enum"
nullable="false" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:choice>
            <xsd:element name="UKaddress"
type="egif:UKAddressStructure"/>
            <xsd:element name="InternationalAddress"
type="egif:InternationalAddressStructure"/>
        </xsd:choice>
    </xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="AQ_Enum">
    <xsd:annotation>
        <xsd:documentation>A Change of Address specific data
type</xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="Residential_Address"/>
        <xsd:enumeration value="Correspondence_Address"/>
        <xsd:enumeration value="Self_Employed_Business_Address"/>
        <xsd:enumeration value="Business_Address"/>
    </xsd:restriction>
</xsd:simpleType>
</xsd:schema>

```

3.2.2. Address Types

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- Schema for commonly used address types -->
<!-- edited with XML Spy v3.5 NT (http://www.xmlspy.com) by Adrian Kent
(private) -->
<xsd:schema targetNamespace="http://GovTalk/Schemas/ChangeOfAddress00d.xsd"
xmlns:xsd="http://www.w3.org/2000/10/XMLSchema"
xmlns:egif="http://GovTalk/Schemas/ChangeOfAddress00d.xsd"
elementFormDefault="qualified" attributeFormDefault="unqualified">
    <xsd:include schemaLocation="BS7666Schema3.xsd"/>
    <xsd:include schemaLocation="SimpleTypes.xsd"/>
    <!-- start of InternationalAddress type definition -->
    <xsd:complexType name="InternationalAddressStructure">
        <xsd:sequence>
            <xsd:element name="IntAddressLine"
type="egif:AddressLineType" minOccurs="2" maxOccurs="5"/>
            <xsd:element name="Country" type="egif:AddressLineType"
default="UK" minOccurs="0"/>
            <xsd:element name="InternationalPostCode"
type="egif:InternationalPostCodeType" minOccurs="0"/>
        </xsd:sequence>
    </xsd:complexType>
    <!-- end of InternationalAddress type definition -->
    <!-- start of AddressUKpostalStructure type definition -->
    <xsd:complexType name="UKPostalAddressStructure">
        <xsd:sequence>
            <xsd:element name="Line" type="egif:AddressLineType"
minOccurs="2" maxOccurs="5"/>
            <xsd:element name="PostCode" type="egif:PostCodeUKtype"
minOccurs="0"/>
        </xsd:sequence>
    </xsd:complexType>
    <!-- end of AddressUKpostalStructure type definition -->
    <!-- start of AddressUKstructure type definition -->

```

```

    <xsd:complexType name="UKAddressStructure">
      <xsd:annotation>
        <xsd:documentation>Supports BS766 address
types.</xsd:documentation>
      </xsd:annotation>
      <xsd:sequence>
        <xsd:choice>
          <xsd:element name="BS7666Address"
type="egif:BSaddressStructure"/>
          <xsd:element name="A_5LineAddress"
type="egif:UKPostalAddressStructure"/>
        </xsd:choice>
        <xsd:element name="UniquePropertyReferenceNumber"
type="egif:UniquePropertyReferenceNumberType" minOccurs="0"/>
        <xsd:choice>
          <xsd:element name="SortCode"
type="egif:MailSortCodeType" minOccurs="0"/>
          <xsd:element name="WalkSort"
type="egif:WalkSortCodeType" minOccurs="0"/>
        </xsd:choice>
      </xsd:sequence>
    </xsd:complexType>
    <!-- end of AddressUKstructure type definition -->
    <!-- start of TelephoneStructure type definition -->
    <xsd:complexType name="TelephoneStructure">
      <xsd:sequence>
        <xsd:element name="TelNationaNumber"
type="egif:TelephoneNumberType"/>
        <xsd:element name="TelExtensionNumber"
type="egif:TelephoneExtensionType" minOccurs="0"/>
        <xsd:element name="TelCountryCode"
type="egif:TelCountryCodeType" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="TelUse" type="egif:WorkHomeType"/>
      <xsd:attribute name="TelMobile" type="egif:YesNoType"/>
      <xsd:attribute name="TelPreferred" type="egif:YesNoType"/>
    </xsd:complexType>
    <!-- end of TelephoneStructure type definition -->
    <!-- start of FaxStructure type definition -->
    <xsd:complexType name="FaxStructure">
      <xsd:sequence>
        <xsd:element name="FaxNationaNumber"
type="egif:TelephoneNumberType"/>
        <xsd:element name="FaxExtensionNumber"
type="egif:TelephoneExtensionType" minOccurs="0"/>
        <xsd:element name="FaxCountryCode"
type="egif:TelCountryCodeType" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="FaxUse" type="egif:WorkHomeType"/>
      <xsd:attribute name="FaxMobile" type="egif:YesNoType"/>
      <xsd:attribute name="FaxPreferred" type="egif:YesNoType"/>
    </xsd:complexType>
    <!-- end of FaxStructure type definition -->
    <!-- start of EmailStructure type definition -->
    <xsd:complexType name="EmailStructure">
      <xsd:sequence>
        <xsd:element name="EMailAddress"
type="egif:RestrictedStringType"/>
      </xsd:sequence>
      <xsd:attribute name="EmailUsage" type="egif:WorkHomeType"/>
      <xsd:attribute name="EMailPreferred" type="egif:YesNoType"/>

```

```

</xsd:complexType>
<!-- end of EmailStructure type definition -->
<xsd:simpleType name="AddressLineType">
  <xsd:restriction base="egif:RestrictedStringType">
    <xsd:minLength value="1"/>
    <xsd:maxLength value="35"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="PostCodeUKtype">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[A-Z]{1,2}[0-9R][0-9A-Z]?[0-9][A-
Z]{2}"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="MailSortCodeType">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[0-9]{5}"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="WalkSortCodeType">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[0-9]{8}"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="UniquePropertyReferenceNumberType">
  <xsd:restriction base="xsd:int">
    <xsd:pattern value="[0-9]{1,12}"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="InternationalPostCodeType">
  <xsd:restriction base="egif:RestrictedStringType">
    <xsd:minLength value="1"/>
    <xsd:maxLength value="35"/>
  </xsd:restriction>
</xsd:simpleType>
</xsd:schema>

```

3.2.3. BS7666 Address Types

```

<?xml version="1.0"?>
<!-- edited with XML Spy v3.5 NT (http://www.xmlspy.com) by Adrian Kent
(private) -->
<xsd:schema targetNamespace="http://GovTalk/Schemas/ChangeOfAddress00d.xsd"
xmlns:xsd="http://www.w3.org/2000/10/XMLSchema"
xmlns:bs7666="http://GovTalk/Schemas/ChangeOfAddress00d.xsd">
  <xsd:annotation>
    <xsd:documentation>BS7666 Schema Definition Purpose: Definition
of basic BS7666 types and constructs. Includes a definition of a basic
BS7666 Address Not Full Schema Date: 05/01/01</xsd:documentation>
  </xsd:annotation>
  <!-- Basic BS7666 Compliant Address Element -->
  <xsd:complexType name="BSaddressStructure">
    <xsd:annotation>
      <xsd:documentation>BS7666 Compliant Address
Structure</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="SAON" type="bs7666:SAONtype"
minOccurs="0">
        <xsd:annotation>

```

```

Object</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
  <xsd:element name="PAON" type="bs7666:PAONtype">
    <xsd:annotation>
      <xsd:documentation>Primary Addressable
Object</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
  <xsd:element name="StreetDescription"
type="bs7666:StreetDescriptorType">
    <xsd:annotation>
      <xsd:documentation>Street
Description</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
  <xsd:choice>
    <xsd:annotation>
      <xsd:documentation>At least one of Town,
County and Locality</xsd:documentation>
    </xsd:annotation>
    <xsd:choice>
      <xsd:annotation>
        <xsd:documentation>One
of</xsd:documentation>
      </xsd:annotation>
      <xsd:element ref="bs7666:Town"/>
      <xsd:element ref="bs7666:Locality"/>
      <xsd:element ref="bs7666:County"/>
    </xsd:choice>
    <xsd:sequence>
      <xsd:annotation>

<xsd:documentation>Both</xsd:documentation>
      </xsd:annotation>
      <xsd:element ref="bs7666:Town"/>
      <xsd:element ref="bs7666:Locality"/>
    </xsd:sequence>
    <xsd:sequence>
      <xsd:annotation>

<xsd:documentation>Both</xsd:documentation>
      </xsd:annotation>
      <xsd:element ref="bs7666:Town"/>
      <xsd:element ref="bs7666:County"/>
    </xsd:sequence>
    <xsd:sequence>
      <xsd:annotation>

<xsd:documentation>Both</xsd:documentation>
      </xsd:annotation>
      <xsd:element ref="bs7666:Locality"/>
      <xsd:element ref="bs7666:County"/>
    </xsd:sequence>
    <xsd:sequence>
      <xsd:annotation>

<xsd:documentation>All</xsd:documentation>
      </xsd:annotation>
      <xsd:element ref="bs7666:Town"/>

```

```

        <xsd:element ref="bs7666:Locality"/>
        <xsd:element ref="bs7666:County"/>
    </xsd:sequence>
</xsd:choice>
<xsd:element name="Posttown" type="bs7666:PosttownType"
minOccurs="0"/>
    <xsd:element name="Postcode" type="bs7666:PostcodeType"
minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="UPRN" type="bs7666:UPRNtype"/>
</xsd:complexType>
<!-- UPRN Definition -->
<xsd:simpleType name="UPRNtype">
    <xsd:restriction base="xsd:integer">
        <xsd:minInclusive value="1"/>
        <xsd:maxInclusive value="999999999999"/>
    </xsd:restriction>
</xsd:simpleType>
<!-- USRN Definition -->
<xsd:simpleType name="USRNtype">
    <xsd:restriction base="xsd:integer">
        <xsd:minInclusive value="1"/>
        <xsd:maxInclusive value="999999999"/>
    </xsd:restriction>
</xsd:simpleType>
<!-- PostCode Definition -->
<xsd:simpleType name="PostcodeType">
    <xsd:restriction base="xsd:string">
        <xsd:pattern value="[A-Z]{1,2}[0-9R][0-9A-Z]? [0-9][A-
Z]{2}"/>
    </xsd:restriction>
</xsd:simpleType>
<!-- PAON Definition -->
<xsd:simpleType name="PAONtype">
    <xsd:restriction base="xsd:string">
        <xsd:pattern value="[\d ]{4}[\p{L} ]{4}[\p{L}
][\p{L} #x26-#x29]{1,90}"/>
    </xsd:restriction>
</xsd:simpleType>
<!-- SAON Definition -->
<xsd:simpleType name="SAONtype">
    <xsd:restriction base="xsd:string">
        <xsd:pattern value="[\d ]{4}[\p{L} ]{4}[\p{L}
][\p{L} #x26-#x29]{1,90}"/>
    </xsd:restriction>
</xsd:simpleType>
<!-- Post Town Definition -->
<xsd:simpleType name="PosttownType">
    <xsd:restriction base="xsd:string">
        <xsd:minLength value="0"/>
        <xsd:maxLength value="30"/>
    </xsd:restriction>
</xsd:simpleType>
<!-- Locality Definition -->
<xsd:simpleType name="LocalityType">
    <xsd:restriction base="xsd:string">
        <xsd:minLength value="0"/>
        <xsd:maxLength value="35"/>
    </xsd:restriction>
</xsd:simpleType>
<!-- County Definition -->

```



```

<xsd:simpleType name="CountyType">
  <xsd:restriction base="xsd:string">
    <xsd:minLength value="0"/>
    <xsd:maxLength value="30"/>
  </xsd:restriction>
</xsd:simpleType>
<!-- Town Definition -->
<xsd:simpleType name="TownType">
  <xsd:restriction base="xsd:string">
    <xsd:minLength value="0"/>
    <xsd:maxLength value="30"/>
  </xsd:restriction>
</xsd:simpleType>
<!-- Street Descriptor -->
<xsd:simpleType name="StreetDescriptorType">
  <xsd:restriction base="xsd:string">
    <xsd:minLength value="0"/>
    <xsd:maxLength value="100"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:element name="Town" type="bs7666:TownType"/>
<xsd:element name="Locality" type="bs7666:LocalityType"/>
<xsd:element name="County" type="bs7666:CountyType"/>
</xsd:schema>

```

3.2.4. Citizen related structures

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- Schema for citizen details -->
<!-- edited with XML Spy v3.5 NT (http://www.xmlspy.com) by Adrian Kent
(private) -->
<xsd:schema targetNamespace="http://GovTalk/Schemas/ChangeOfAddress00d.xsd"
xmlns:xsd="http://www.w3.org/2000/10/XMLSchema"
xmlns:egif="http://GovTalk/Schemas/ChangeOfAddress00d.xsd"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xsd:include schemaLocation="SimpleTypes.xsd"/>
  <xsd:include schemaLocation="Address_types.xsd"/>
  <!-- start of CitizenNameStructure type definition -->
  <xsd:complexType name="CitizenNameStructure">
    <xsd:sequence>
      <xsd:element name="CitizenNameTitle"
type="egif:CitizenNameTitleType" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="CitizenNameForename"
type="egif:CitizenNameForenameType" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="CitizenNameSurname"
type="egif:CitizenNameSurnameType"/>
      <xsd:element name="CitizenNameSuffix"
type="egif:CitizenNameSuffixType" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="CitizenNameRequestedName"
type="egif:CitizenNameRequestedNameType" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
  <!-- end of CitizenNameStructure type definition -->
  <!-- start of CitizenRegistrationStructure type definition -->
  <xsd:complexType name="CitizenRegistrationStructure">
    <xsd:sequence>
      <xsd:element name="NationalInsuranceNumber"
type="egif:NationalInsuranceNumberType" minOccurs="0"/>
      <xsd:element name="UniqueTaxReference"
type="egif:UniqueTaxReferenceType" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

```

```

        <xsd:element name="DrivingLicenceNumber"
type="egif:DrivingLicenceNumberType" minOccurs="0"/>
        <xsd:element name="NationalHealthNumber"
type="egif:NationalHealthNumberType" minOccurs="0"/>
        <xsd:choice minOccurs="0">
            <xsd:element name="PassportNumber_Old"
type="egif:PassportNumber_OldType"/>
            <xsd:element name="PassportNumber_New"
type="egif:PassportNumber_NewType"/>
        </xsd:choice>
        <xsd:element name="ElectoralRollNumber"
type="egif:ElectoralRollNumberType" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>
<!-- end of CitizenRegistrationStructure type definition -->
<!-- start of CitizenDetailsStructure type definition -->
<xsd:complexType name="CitizenDetailsStructure">
    <xsd:sequence>
        <xsd:element name="CitizenName"
type="egif:CitizenNameStructure"/>
        <xsd:element name="CitizenRegistration"
type="egif:CitizenRegistrationStructure"/>
        <xsd:element name="PreferredLanguages"
type="egif:LangCode" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="ContactDetails"
type="egif:CitizenContactDetailsStructure"/>
        <xsd:element name="CitizenSex" type="egif:SexTypeCode"
minOccurs="0"/>
        <xsd:element name="CitizenBirthDate"
type="egif:CitizenBirthDateStructure" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>
<!-- end of CitizenDetailsStructure type definition -->
<!-- start of CitizenContactDetailsStructure type definition -->
<xsd:complexType name="CitizenContactDetailsStructure">
    <xsd:sequence>
        <xsd:element name="Email" type="egif:EmailStructure"
minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="Telephone"
type="egif:TelephoneStructure" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="Fax" type="egif:FaxStructure"
minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
</xsd:complexType>
<!-- end of CitizenContactDetailsStructure type definition -->
<!-- start of CitizenBirthDateStructure type definition -->
<xsd:complexType name="CitizenBirthDateStructure">
    <xsd:sequence>
        <xsd:annotation>
            <xsd:documentation>This might change to allow codes
as well as descriptions of for verification</xsd:documentation>
        </xsd:annotation>
        <xsd:element name="BirthDate" type="egif:DateType"/>
        <xsd:element name="VerifiedBy"
type="egif:VerifiedByType"/>
    </xsd:sequence>
</xsd:complexType>
<!-- end of CitizenBirthDateStructure type definition -->
<xsd:simpleType name="VerifiedByType">
    <xsd:annotation>

```

```

        <xsd:documentation>A Change of Address specific data
type</xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="not verified"/>
        <xsd:enumeration value="accepted on balance of
probabilities"/>
        <xsd:enumeration value="secondary certificate"/>
        <xsd:enumeration value="certified copy of birth
certificate"/>
        <xsd:enumeration value="short form birth certificate or
certificate of registration of birth"/>
        <xsd:enumeration value="birth certificate"/>
    </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="CitizenNameTitleType">
    <xsd:restriction base="egif:RestrictedStringType">
        <xsd:minLength value="1"/>
        <xsd:maxLength value="35"/>
    </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="CitizenNameForenameType">
    <xsd:restriction base="egif:RestrictedStringType">
        <xsd:minLength value="1"/>
        <xsd:maxLength value="35"/>
    </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="CitizenNameSurnameType">
    <xsd:restriction base="egif:RestrictedStringType">
        <xsd:minLength value="1"/>
        <xsd:maxLength value="35"/>
    </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="CitizenNameSuffixType">
    <xsd:restriction base="egif:RestrictedStringType">
        <xsd:minLength value="1"/>
        <xsd:maxLength value="35"/>
    </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="CitizenNameRequestedNameType">
    <xsd:restriction base="egif:RestrictedStringType">
        <xsd:minLength value="1"/>
        <xsd:maxLength value="70"/>
    </xsd:restriction>
</xsd:simpleType>
</xsd:schema>

```

3.2.5. Simple types

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v3.5 NT (http://www.xmlspy.com) by Adrian Kent
(private) -->
<xsd:schema targetNamespace="http://GovTalk/Schemas/ChangeOfAddress00d.xsd"
xmlns:egif="http://GovTalk/Schemas/ChangeOfAddress00d.xsd"
xmlns:xsd="http://www.w3.org/2000/10/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
    <!--=====-->
    <!-- Common Simple Types -->
    <!--=====-->
    <xsd:simpleType name="PopulatedStringType">

```

```

        <xsd:restriction base="xsd:string">
            <xsd:minLength value="1"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="RestrictedStringType">
        <xsd:restriction base="xsd:string">
            <xsd:pattern value="[A-Za-z0-9!&quot;#%&apos;()*+,-
./:;&lt;=&gt;?@[\\&#093;^_`{|}~]*"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="DateType">
        <xsd:annotation>
            <xsd:documentation>A Common DataType W 3C XML Schema
datatype</xsd:documentation>
        </xsd:annotation>
        <xsd:restriction base="xsd:date"/>
    </xsd:simpleType>
    <xsd:simpleType name="YesNoType">
        <xsd:annotation>
            <xsd:documentation>A Common DataType</xsd:documentation>
        </xsd:annotation>
        <xsd:restriction base="xsd:string">
            <xsd:enumeration value="yes"/>
            <xsd:enumeration value="no"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="SexTypeCode">
        <xsd:annotation>
            <xsd:documentation>A Common DataType</xsd:documentation>
        </xsd:annotation>
        <xsd:restriction base="xsd:string">
            <xsd:enumeration value="male"/>
            <xsd:enumeration value="female"/>
            <xsd:enumeration value="unknown"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="LangCode">
        <xsd:annotation>
            <xsd:documentation>A Common DataType Values from the ISO
standard for 2 character language codes (ISO 639:1988 Code for the
representation of names of languages). </xsd:documentation>
        </xsd:annotation>
        <xsd:restriction base="xsd:string">
            <xsd:enumeration value="en"/>
            <xsd:enumeration value="cy"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="NationalInsuranceNumberType">
        <xsd:restriction base="xsd:string">
            <xsd:pattern value="[A-CEGHJ-NPR-TW-Z]{2}[0-
9]{6}[ABCD\S]{1}"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="UniqueTaxReferenceType">
        <xsd:restriction base="xsd:string">
            <xsd:pattern value="[0-9]{1,10}"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="PassportNumber_OldType">
        <xsd:restriction base="egif:PopulatedStringType">
            <xsd:pattern value="[A-Z,0-9]{1}[0-9]{4}[A-Z,0-9]{1}"/>

```

```

        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="PassportNumber_NewType">
        <xsd:restriction base="xsd:int">
            <xsd:pattern value="[0-9]{9}" />
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="DrivingLicenceNumberType">
        <xsd:restriction base="egif:PopulatedStringType" />
    </xsd:simpleType>
    <xsd:simpleType name="NationalHealthNumberType">
        <xsd:restriction base="egif:PopulatedStringType" />
    </xsd:simpleType>
    <xsd:simpleType name="ElectoralRollNumberType">
        <xsd:annotation>
            <xsd:documentation>No Pattern yet for Electoral roll
number</xsd:documentation>
        </xsd:annotation>
        <xsd:restriction base="egif:PopulatedStringType" />
    </xsd:simpleType>
    <xsd:simpleType name="EmailAddressType">
        <xsd:restriction base="xsd:string">
            <xsd:minLength value="1" />
            <xsd:maxLength value="254" />
            <xsd:pattern value="[0-9A-Za-z\.\_\-]{1,127}@[0-9A-Za-
z\.\_\-]{1,127}" />
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="TelCountryCodeType">
        <xsd:restriction base="xsd:string">
            <xsd:pattern value="[0-9]{1,3}" />
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="TelephoneNumberType">
        <xsd:restriction base="xsd:string" />
    </xsd:simpleType>
    <xsd:simpleType name="TelephoneExtensionType">
        <xsd:restriction base="xsd:string">
            <xsd:pattern value="[0-9]{1,6}" />
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="WorkHomeType">
        <xsd:annotation>
            <xsd:documentation>A Change of Address specific data
type</xsd:documentation>
        </xsd:annotation>
        <xsd:restriction base="xsd:string">
            <xsd:enumeration value="work" />
            <xsd:enumeration value="home" />
        </xsd:restriction>
    </xsd:simpleType>
</xsd:schema>

```

4. Approval

5. Abbreviations

Abb	Definition
DSC	Data Standards Catalogue
eGIF	Electronic Government Interoperability Framework
FTP	File Transfer Protocol (an Internet protocol for managing and transferring files)
HTTP	Hypertext Transfer Protocol
OeE	Office of the E-Envoy (a part of the Cabinet Office)
OMG	Object management Group
UML	Unified Modelling Language
URL	Uniform Resource Locator
W3C	World Wide Web Consortium
XML	Extensible Markup Language
XSDL	XML Schema Definition Language

6. References

Document References	
[1]	The eGovernment Interoperability Framework
[2]	UK Online – Information Architecture - Overview, draft 0.1a, 28 July 2000
[3]	OMG Unified Modelling Language Specification, Version 1.3, June 1999 ftp://ftp.omg.org/pub/docs/ad/99-06-08.pdf
[4]	XML Schema Part 0: Primer, W3C Working Draft, 7 April 2000 http://www.w3.org/TR/xmlschema-0/
[5]	XML Schema Part 1: Structures, W3C Working Draft, 7 April 2000 http://www.w3.org/TR/xmlschema-1/
[6]	XML Schema Part 2: Datatypes, W3C Working Draft, 7 April 2000 http://www.w3.org/TR/xmlschema-2/
[7]	Data Standards Catalogue, Version 0.3, March 2001
[8]	

END OF DOCUMENT