



ONIX Structured Name & Address Model Format Overview

DRAFT Version 0.122

May 2013

Incorporating changes arising during schema generation and internal review

Last updated (TD) 20-May-13

Please note that this document is a working draft.
EDItEUR invites comments on this specification and the associated XML schema. Please send any comments, suggestions for improvement, or questions to info@editeur.org.

Changes in versions 0.12, 0.121 & 0.122

This version incorporates a series of changes and improvements arising from an internal EDItEUR review during schema generation:

- A number of data elements have been renamed for consistency and greater clarity.
- In particular, those composites containing identifiers have been renamed in the style xxxxIdentifier while two of the element types they contain are renamed to simply IDTypeName and IDValue, as in this example:
 - <PersonIdentifier>
 - <PersonIDType>
 - <IDTypeName>
 - <IDValue>
- Elements containing textual rather than controlled-value names have been renamed to have a Name suffix, as in CountryName.
- Some composite definitions have been elaborated, so that it is clearer which elements or combinations thereof are expected.
- Code list additions have been proposed to allow controlled-value population of various elements.
- The relationship between the original <StructuredAddress> composite and that for <AddressLine> has been modified. The StructuredAddress wrapper has been removed and all the elements it contains become “child elements” of <PostalAddress>, alongside the repeatable <AddressLine>.
- <CountrySubdivision> is now a repeatable composite.

- The element <KeyNames> is now mandatory within the <PersonNameStructured> composite.
- Various other textual and descriptive changes have been made to improve clarity and explanation.

Copyright © 2013 EDItEUR Limited. All rights reserved.

EDItEUR is the international group that coordinates the development and promotion of standards for electronic commerce in the books and serials sectors.

1. Introduction

This document describes a Structured Name & Address model, developed by EDItEUR in conjunction with a specialist Working Group of ICEDIS, the International Committee on EDI for Serials. The model provides a highly structured means of communicating names, addresses and party identifiers between various players active in the library supply chain. The model is expressed in XML and is intended for use within the ONIX, ICEDIS and EDItX messages developed by EDItEUR.

Recognizing the parties involved, quickly and unambiguously, is a key first step in any effective supply chain. The current work aims to support automated, machine-to-machine communication of name and address information and thus to enable automated matching and processing of the business messages concerned. The challenge is to supplant the many forms of unstructured or free-text communications commonly encountered, which frequently necessitate manual keying or other intervention.

The model presented here will initially be used for business exchanges between parties in the library supply chain. But its structure is intended to be general and extensible enough that it can in future be incorporated in other EDItEUR standard formats as the need arises.

2. Business drivers

Transactional messages such as EDItX used in electronic trading require the ability to carry names and addresses, frequently those of organizations and sometimes those of persons. ONIX product information formats require the ability to carry personal and organizational names and (sometimes) addresses. EDItEUR therefore has a general interest in developing a robust and extensible framework for carrying structured party information. The current ICEDIS requirement for structured communication of names and addresses, in conjunction with a message for new orders, provided the impetus for the first phase of development.

Analysis of the ways in which names and addresses are used in these different contexts reveals a number of different but substantially overlapping requirements:

- a. Most electronic trading messages are sent between companies (using the word “company” here also to mean any organizational body). If addresses are included, they refer to locations belonging to a company. If personal names are included, they refer to persons within the company who may be contacted in connection with a particular transaction or a particular type of activity. In other words, the approach and natural logic of the required data structure are *company-based*.
- b. However, in a “consumer direct fulfilment” or “drop ship” business, the buyer and supplier are both companies but the delivery address is most often that of a private person. So even in company-to-company trading there is sometimes a need for a *person-based* approach.
- c. In ONIX product information messages, persons appear as contributors to the content of a publication, but in some cases their names may need to be accompanied by their professional affiliation and/or by addresses or other contact detail that may be personal or based on their company affiliation. So we need a *person-based* approach that is *capable of carrying company or other organizational information*.
- d. In an ONIX message describing a subscription product, the publisher of the product may be identified along with a whole series of different contacts for different purposes,

each of which may involve persons or companies or both – e.g. the advertising contact may be a person at a third-party agency. So we need a *company-based* approach that is *capable of carrying both person and other company information*.

For the reasons outlined above, EDItEUR has developed a flexible model allowing the data to be structured from either Person-based or Organization-based perspectives, as outlined in Section 9 below.

3. Party identifiers

Alongside the structured representation of names and addresses, party identifiers offer an effective and complementary method for recognizing and differentiating between customers (and other players) in the supply chain. For this reason, the model offers straightforward but comprehensive facilities for conveying one or more identifiers for each of the parties involved. A small series of elements carries information on which identifier is in use as well as the identifier itself. The model includes controlled value code lists to declare those identifiers most commonly encountered.

Proprietary identifiers such as those assigned by agents or publishers are widely used in business-to-business communications. A number of public standard identifiers are also relevant here, including the [DUNS](#) (Dun & Bradstreet Data Universal Numbering System), [GLN](#) (Global Location Number), [ISNI](#) (International Standard Name Identifier), [ORCID](#) (Open Researcher and Contributor ID), [Ringgold ID](#) and [SAN](#) (Standard Address Number). These and other identifiers that have significant uptake in the library supply chain are included in the code lists used by the model.

Users of the model are strongly encouraged to include party identifiers understood by both sender and recipients whenever these are available.

4. Geographical place identifiers

Controlled identifiers for geographical entities offer similar advantages to those of party identifiers when used in conjunction with the model. The greater the degree to which addresses can be expressed in standardized forms, the lesser the opportunity for error or misinterpretation. The model is designed to convey a number of publically available codes and in some instances supports both literal and coded representations (e.g. “The Netherlands” and “NL”).

Alongside the GLN (Global Location Number), at least four levels can be distinguished, represented by the postal or zip code, the postal town or city, the country subdivision and the country itself. The ISO 3166-2 standard covers country subdivisions and countries, whilst the [UN/LOCODE](#) database offers a useful resource that incorporates ISO 3166-2 and takes coded coverage to the level of individual towns in many countries of the world. Each of these standards (and others) can now be referenced within the appropriate <xxxxIdentifier> composites used within the model.

5. Communication & matching “versus” presentation

It is important to emphasize that the primary purpose of the Structured Name & Address model is to provide a vehicle for communicating name and address and other party information. This is potentially quite different from some of the other uses to which the

names and addresses may be put, such as conforming to the presentational requirements of mailing labels or the postal recommendations of particular countries.

The role of the model is to carry names and addresses in segments or elements that can be universally understood and interpreted. Other processes may well be necessary in sending or, particularly, receiving systems to store information from the model in the appropriate database fields, to build or present information in ways appropriate for particular applications, or to concatenate or truncate the data for local usage.

A key advantage of a very structured representation, such as the EDItEUR model, is that it is easier to carry out matching between incoming records and those in receiving (or indeed other) databases. This contrasts markedly with situations in which the incoming data is of a “free text”, uncontrolled nature – the latter posing great challenges in interpretation and increasing risks of mismatching or simple non-recognition of the information.

6. Use case scenarios envisaged

The structured name & address model is not a message in itself; it can better be thought of as a module that can be utilized in a range of situations where the requirement exists to exchange names and addresses.

For this reason, it was felt appropriate to pilot the model in conjunction with a message currently under development, namely that for New Orders for subscription products. A major reason that new orders have to be processed manually at present is that the lack of structured addresses makes automated customer identification difficult if not impossible. Therefore the model will be used to convey name and address information as part of the New Order message.

Once pilots are successfully completed, however, it is envisaged that the model will be suitable for deployment in a number of other areas. In the library supply chain alone, these are likely to include the following:

- To convey name and address information as part of Renewal (as opposed to New) orders for subscription products.
- To convey name and address information as part of address change notifications.
- To synchronize, correct and update customer records between agents and publishers ahead of key operations such as renewals or invoicing, in order to minimize errors and maximize auto-processing percentages.
- To help uniquely identify customers as part of transfer notifications, for instance when a subscription has been transferred between agents or journals have been transferred between publishers.
- To better identify a customer as a prelude to pricing negotiations or creating bespoke deals.

7. About the business purpose of a particular address

The names and addresses carried by the model will each be associated with a certain business purpose – such as a delivery address, a billing or accounts department, a consolidation location, etc. After discussion, the Working Group decided that it would be most appropriate to declare the purpose of each address by the context in which the model

is used. In other words, there is no “purpose” code within the model itself: the purpose will be made clear within the “parent” message.

8. About special characters, languages and scripts

Names and addresses can of course contain not only simple, unaccented characters from European languages using the latin alphabet. Accents, other diacritical marks or special characters are routinely encountered, particular in the international market for subscription products. The model proposed supports this requirement from the outset, by specifying the appropriate character set (likely to be UTF-8) for any messages that include the model.

Whilst message development and piloting are underway, the model will be configured to focus on English-language communication. Later, however, it will be possible straightforwardly to add either language or script attributes to particular elements, if business requirements so dictate; this would allow the model natively to accommodate languages that are not western European in origin as well as those that utilize different scripts entirely.

9. A high-level view of the model

The model may be represented in outline as follows. Note that the “entry” point chosen, whether Person-based or Company-based, is essentially a business decision driven by the context within which the model is used.

Person-based

Person	The whole structure is contained within a <Person> element
PersonIdentifier	Person ID codes, name codes and names
PersonCommunicationDetails	Communication details that apply to a person independent of address
PersonLocationDetails	Postal address, communication details at that address
AssociatedOrganization	Organization associated with the person concerned
AssociationType	For example, professional affiliation
Organization	Nested <Organization> element, described below
AssociatedPerson	Other person associated with the person concerned
AssociationType	For example, personal assistant
Person	Nested <Person> element

Organization-based

Organization	The whole structure is contained within an <Organization> element
OrganizationIdentifier	Organization ID codes, name codes and names
OrganizationCommunication Details	Communication details for an organization independent of address
OrganizationLocationDetails	Location codes, postal address, communication details at that address
AssociatedOrganization	Other organization associated with the organization concerned
AssociationType	For example, distributor
Organization	Nested <Organization> element
AssociatedPerson	Person associated with the organization concerned
AssociationType	For example, sales representative
Person	Nested <Person> element, described above

Subsequent pages present a draft version of how the overall model is structured in terms of individual data elements. Details of the cardinalities of each element (whether it is optional

or mandatory and whether or not it is repeatable) and some of the code lists to be used are included for discussion and confirmation.

For the immediate purposes of the ICEDIS Working Group, only a subset of the data elements will be implemented in the current phase. Other elements in the full model are retained here for illustration, but are greyed out **THUS** in the following tables.

The data model in detail: Person-based

Proposed cardinalities are shown in the right-hand column for each element: 1 = Mandatory, not repeatable within its parent. 0-1 = Optional, not repeatable within its parent. 1-n = Mandatory, repeatable within its parent. 0-n = Optional, repeatable within its parent.

Elements greyed out **THUS** are retained as placeholders for future extensions of the model.

Person	The whole structure is contained within a <Person> element. Must contain at least one instance (or more) of <PersonIdentifier> and <Name>	
PersonIdentifier	A coded identification of the public identity or persona of a natural person, such as an ORCID identifier	0-n
PersonIDType	A code indicating the scheme from which a person ID code is taken. See code list 44E for permissible values	1
IDTypeName	Only appears if <PersonIDType> indicates a proprietary coding scheme	0-1
IDValue	Mandatory if <PersonIdentifier> is used	1
Name	<Name> includes information on types of names as well as the names themselves. Must contain at least one (or more) of <PersonName>, <PersonNameInverted> or <PersonNameStructured>	0-n
NameType	A code indicating the type of the name (eg pseudonym). See code list 18 for permissible values: default = "00" (unspecified)	1
PersonName	A person name as a single text string in normal order of presentation	0-1
PersonNameInverted	A person name as a single text string, inverted so that a surname or other "key" name appears first	0-1
PersonNameStructured	A person name structured in accordance with ONIX conventions. See explanations below	0-1
TitlesBeforeNames	The first part of a structured name of a person: qualifications and/or titles preceding a person's names, e.g. <i>Professor</i> or <i>HRH Prince</i> or <i>Saint</i>	0-1
NamesBeforeKey	The second part of a structured name of a person: name(s) and/or initial(s) preceding a person's key name(s), e.g. <i>James J.</i>	0-1
PrefixToKey	The third part of a structured name of a person: a prefix which precedes the key name(s) but which is not to be treated as part of the key name, e.g. <i>van</i> in <i>Ludwig van Beethoven</i>	0-1

KeyNames	The fourth part of a structured name of a person: key name(s), i.e. the name elements normally used to open an entry in an alphabetical list, e.g. <i>Smith</i> or <i>Garcia Marquez</i>	1
NamesAfterKey	The fifth part of a structured name of a person: name suffix, or name(s) following a person's key name(s), e.g. <i>Ibrahim</i> (in <i>Anwar Ibrahim</i>)	0-1
SuffixToKey	The sixth part of a structured name of a person: a suffix following a person's key name(s), e.g. <i>Jr</i> or <i>III</i>	0-1
LettersAfterNames	The seventh part of a structured name of a person: qualifications or honours following a person's names, e.g. <i>CBE</i> or <i>FRS</i>	0-1
TitlesAfterNames	The eighth part of a structured name of a person: titles following a person's names, e.g. <i>Duke of Edinburgh</i>	0-1
PersonCommunicationDetails	Communication details that apply to a person	0-n
CommunicationType	A type of communication method, eg phone or email. See code list 205 for permissible values	1
CommunicationLocator	Telephone number, email address, URL etc	1
PersonLocationDetails	Must include either at least one instance of <LocationIdentifier> and <PostalAddress> or both	0-n
LocationType	A code identifying a type of personal location, eg home. See code list 206 for permissible values	0-1
LocationIdentifier	A coded identification of a location. This is repeatable, to allow multiple identifiers for the same location to be included if available	0-n
LocationIDType	A code indicating the scheme from which a location code is taken, eg EAN-UCC GLN, book trade SAN. See code list 44 for permissible values	1
IDTypeName	Only appears if <LocationIDType> indicates a proprietary coding scheme	0-1
IDValue		1
PostalAddress	The <PostalAddress> may be presented entirely as a series of structured elements (strongly recommended) or may also include one or a number of free-text <AddressLine>s if the fully structured approach is not feasible for the sender of the message	0-1
AddressLine	Repeatable free text string for lines of address. This option is included here for cases where structured information is not available to the sender of the message	0-n
PostalServiceBoxID		0-1

StreetLocation		0-n
StreetLocationName	The name of a street, an estate, or a group of buildings, eg Chertsey Road, Acme Industrial Estate, Peabody Buildings	0-1
NumberWithinStreetLocation		0-1
NameWithinStreetLocation	The name of a house or building, eg Globe House, Dunroamin	0-1
BlockOrEntrance	The identification of a block or entrance within a building or group of buildings, eg Portal B	0-1
Floor		0-1
Apartment		0-1
DistrictName	For example, village or neighbourhood	0-n
PostalTownCityName	The name of a postal town or city. Elements from <PostalTownCityName> down to <CountryCoded> are now included within <PostalAddress>	0-1
PostalTownCityIdentifier	A coded identification of a postal town or city. Note that this is not the same as the postal or zip code, which should appear in the element <PostalCode>	0-n
PostalTownCityIDType	A code indicating the scheme from which a postal town or city code is taken, eg UN/LOCODE. See code list 207A for permissible values	1
IDTypeName	Only appears if <PostalTownCityIDType> indicates a proprietary coding scheme	0-1
IDValue		1
PostalCode		0-1
CountrySubdivision	The possibility of allowing a repeatable <CountrySubdivision> composite is introduced here for discussion within the Working Group. Must contain either <CountrySubdivisionName> or <CountrySubdivisionIdentifier> or both	0-n
CountrySubdivisionName	State, province, county	0-1
CountrySubdivisionIdentifier	A coded identification of a country subdivision	0-n
CountrySubdivisionIDType	A code indicating the scheme from which a country subdivision code is taken, eg ISO 3166 country subdivisions. See code list 207B for permissible values	1
IDTypeName	Only if <CountrySubdivisionIDType> indicates a proprietary coding scheme	0-1

IDValue		1
CountryName	Country name	0-1
CountryCode	See code list 91 for permissible values: ISO 3166 two-character country codes	1
LocationCommunicationDetails	Communication details that are specific to a location	0-n
CommunicationType	A type of communication method, eg phone or email. See code list 205 for permissible values	1
CommunicationLocator	Telephone number, email address, URL etc	1
AssociatedPerson	Person named as a contact at a stated personal location	0-n
AssociationType	A code indicating a type of association, eg personal assistant. Could be mandatory but only if we have a code list containing at least a default value, such as "unspecified"	0-1
AssociationDescription	A free text description of the association	0-1
Person	Nested <Person> element	1
AssociatedOrganization	An organizational body associated with the person concerned	0-n
AssociationType	A code indicating a type of association, eg professional affiliation. See code list 208 for permissible values	0-1
AssociationDescription	A free text description of the association, eg "Librarian", "Managing Director", "Hunterian Professor of Mathematics"	0-1
Organization	Nested <Organization> element, documented below	1
AssociatedPerson	Other person associated with the person concerned	0-1
AssociationType	A code indicating a type of association, eg agent. Could be mandatory but only if we have a code list containing at least a default value, such as "unspecified"	0-1
AssociationDescription	A free text description of the association	0-1
Person	Nested <Person> element	1

The data model in detail: Organization-based

Organization	The whole structure is contained within an <Organization> element. Must contain at least one instance (or more) of <OrganizationIdentifier> and <Name>	
OrganizationIdentifier	A coded identification of the public identity of an organizational body, eg a GLN or a Ringgold ID. See code list 44A for permissible values	0-n
OrganizationIDType	A code indicating the scheme from which an organizational ID code is taken. See code list 44A for permissible values	1
IDTypeName	Only appears if <OrganizationIDType> indicates a proprietary coding scheme	0-1
IDValue		1
Name	<Name> includes information on types of names as well as the names themselves. Must contain at least one (or more) of <OrganizationName>, <OrganizationName Inverted> and <OrganizationNameStructured>	0-n
NameType	A code indicating the type of name (eg registered name, trading name). See code list 18 for permissible values: default = "00" (unspecified)	1
OrganizationName	An organizational name as a single text string in normal order of presentation	0-1
OrganizationNameInverted		0-1
OrganizationNameStructured		0-1
PrincipalName		1
SubdivisionName	Division, department etc: repeatable for multiple levels	0-n
OrganizationNameDescription	Free text supplementary to an organizational name, eg "A division of Interstellar Conglomerates"	0-1
OrganizationCommunicationDetails	Communication details that apply to an organizational body independent of location	0-n
CommunicationType	A type of communication method, eg phone or email. See code list 205 for permissible values	1
CommunicationLocator	Telephone number, email address, URL etc	1
OrganizationLocationDetails	Must include either at least one instance of <LocationCoded> or <PostalAddress> or both	0-n

LocationType	A code identifying a type of personal location, eg home. See code list 206 for permissible values	0-1
LocationIdentifier	A coded identification of a location. This is repeatable, to allow multiple identifiers for the same location to be included if available	0-n
LocationIDType	A code indicating the scheme from which a location code is taken, eg EAN-UCC GLN, book trade SAN. See code list 44 for permissible values	1
IDTypeName	Only if <LocationIDType> indicates a proprietary coding scheme	0-1
IDValue		1
PostalAddress	The <PostalAddress> may be presented entirely as a series of structured elements (strongly recommended) or may also include one or a number of free-text <AddressLine>s if the fully structured approach is not feasible for the sender of the message	0-1
AddressLine	Repeatable free text string for lines of address. This option is included here for cases where structured information is not available to the sender of the message	0-n
PostalServiceBoxID		0-1
StreetLocation		0-n
StreetLocationName	The name of a street, an estate, or a group of buildings, eg Chertsey Road, Acme Industrial Estate, Peabody Buildings	0-1
NumberWithinStreetLocation		0-1
NameWithinStreetLocation	The name of a house or building, eg Globe House, Dunroamin	0-1
BlockOrEntrance	The identification of a block or entrance within a building or group of buildings, eg Portal B	0-1
Floor		0-1
Apartment		0-1
DistrictName	For example, village or neighbourhood	0-n
PostalTownCityName	Elements from <PostalTownCityName> down to <CountryCoded> are now included within <PostalAddress>	0-1
PostalTownCityIdentifier	A coded identification of a postal town or city. Note that this is not the same as the postal or zip code, which should appear in the element <PostalCode>	0-n
PostalTownCityIDType	A code indicating the scheme from which a postal town or city code is taken, eg	1

	UN/LOCODE See code list 207A for permissible values	
IDTypeName	Only if <PostalTownCityIDType> indicates a proprietary coding scheme	0-1
IDValue		1
PostalCode		0-1
CountrySubdivision	The possibility of allowing a repeatable <CountrySubdivision> composite is introduced here for discussion within the Working Group	0-n
CountrySubdivisionName	State, province, county	0-1
CountrySubdivisionIdentifier	A coded identification of a country subdivision	0-1
CountrySubdivisionIDType	A code indicating the scheme from which a country subdivision code is taken, eg ISO 3166 country subdivisions. See code list 207B for permissible values	1
IDTypeName	Only if <CountrySubdivisionIDType> indicates a proprietary coding scheme	0-1
IDValue		1
CountryName	Country name	0-1
CountryCode	See code list 91 for permissible values: ISO 3166 two-character country codes	1
LocationCommunicationDetails	Communication details that are specific to a location	0-n
CommunicationType	A type of communication method, eg phone or email. See code list 205 for permissible values	1
CommunicationLocator	Telephone number, email address, URL etc	1
AssociatedPerson	Person named as a contact at a stated organizational location	0-n
AssociationType	A code indicating a type of association, eg professional affiliation. See code list 208 for permissible values	0-1
AssociationDescription	A free text description of the association	0-1
Person	Nested <Person> element. In practice, probably only two components of the <Person> structure would be used here in the context of the New Order message, namely the element <PersonName> and the composite <PersonNameStructured>	1
AssociatedOrganization	Other organization associated with the organizational body	0-n
AssociationType	A code indicating a type of association, eg distributor. Could be mandatory but only if we have a code list containing at least a default value, such as "unspecified"	0-1

AssociationDescription	A free text description of the association	0-1
Organization	Nested <Organization> element	1
AssociatedPerson	Person associated with the organizational body in general, not at a specific organizational location	0-1
AssociationType	A code indicating a type of association, eg agent. Could be mandatory but only if we have a code list containing at least a default value, such as "unspecified"	0-1
AssociationDescription	A free text description of the association	0-1
Person	Nested <Person> element	1