



CIQ TC Specifications

Customer Information Quality Technical Committee

Technical Overview

Version 3.0 (draft)

Date Created: 15 August 2004

Last Updated: 15 December 2005

Editors

Max Voskob, Individual Member, OASIS CIQ TC

Ram Kumar, Individual Member and Chair, OASIS CIQ TC

Contributors

John Glaubitz	Vertex	Member, CIQ TC
Hido Hasimbegovic	Individual	Member, CIQT TC
Robert James	Individual	Member, CIQ TC
Joe Lubenow	Individual	Member, CIQ TC
Mark Meadows	Microsoft Corporation	Member, CIQ TC
John Putman	Individual	Prospective Member, CIQ TC
Michael Roytman	Vertex	Member, CIQ TC
Colin Wallis	New Zealand Government	Member, CIQ TC
David Webber	Individual	Member, CIQ TC

Abstract

This Technical Overview provides a quick practical introduction into high level technical details of CIQ TC specification family version 3.

Intellectual Property Rights, Patents, Licenses and Royalties

CIQ TC Specifications (includes documents, schemas and examples^{1 and 2}) are free of any Intellectual Property Rights, Patents, Licenses or Royalties. Public is free to download and implement the specifications free of charge. Please, read OASIS Copyright Notice in APPENDIX A.

¹xAL-Australia.XML

Address examples come from AS/NZ 4819:2003 standard of Standards Australia and are subject to copyright

²xAL-international.xml

Address examples come from a variety of sources including Universal Postal Union (UPU) website and the UPU address examples are subject to copyright.

Table of Contents

1	INTRODUCTION	4
2	CIQ TC FAMILY VERSION 3.0	5
2.1	THE NEED FOR A NEW VERSION	5
2.2	WHAT IS IN SCOPE IN THIS VERSION	5
2.3	WHAT IS OUT OF SCOPE IN THIS VERSION	5
3	CIQ TC SPECIFICATIONS VERSION 3.0	6
3.1	EXTENSIBLE NAME LANGUAGE (xNL)	6
3.2	EXTENSIBLE ADDRESS LANGUAGE (xAL)	7
3.3	EXTENSIBLE NAME AND ADDRESS LANGUAGE (xNAL) VERSION 3.0	8
3.4	EXTENSIBLE PARTY INFORMATION LANGUAGE (xPIL)	10
3.5	EXTENSIBLE PARTY RELATIONSHIPS LANGUAGE (xPRL)	11
4	PRACTICAL APPLICATIONS OF CIQ TC SPECIFICATIONS	12
4.1	WHERE TO START	12
4.2	DON'T GET CONFUSED – KEEP IT SIMPLE	12
4.3	DATA EXCHANGE	12
4.4	OUTPUT FORMATTING	12
4.5	SCHEMA EXTENSIONS	12
4.6	DATA MAPPING CHALLENGES	13
4.6.1	<i>Application diversity</i>	13
4.6.2	<i>Cultural diversity</i>	13
4.6.3	<i>CIQ TC solution</i>	13
	APPENDIX A. NOTICES	14

1 Introduction

This document is a brief technical overview of version 3.0 of OASIS CIQ TC specifications family namely:

- **xNL** : extensible Name Language
- **xAL**: extensible Address Language
- **xNAL**: extensible Name and Address Language (combines xNL and xAL)
- **xPIL**: extensible Party Information Language (formerly known as extensible Customer Information language (**xCIL**))
- **xPRL**: extensible Party Relationships Language (formerly known as extensible Customer Relationships Language (**xPRL**) – Release data for this specification not set yet

The purpose of this document also is to give software developers and solution architects a quick snapshot of CIQ TC specifications and help decide if the specifications are suitable for a particular application.

Status

This document is currently a draft version and will be updated periodically on no particular schedule. Send comments to the editor.

Committee members should send comments on this specification to the ciq@lists.oasis-open.org list. Others should subscribe to and send comments to the ciq-comment@lists.oasis-open.org list. To subscribe, send an email message to ciq-comment-request@lists.oasis-open.org with the word "subscribe" as the body of the message.

General public may also use "Send comment" option on OASIS CIQ TC home page to submit any feedback.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the CIQ TC web page (<http://www.oasis-open.org/committees/ciq/>).

The errata page for this specification is at <http://www.oasis-open.org/committees/xxx/yyy>.

2 CIQ TC Family Version 3.0

2.1 The need for a new version

The CIQ TC's XML Name and Address languages define universal structures for name and address entities.

It is a trivial exercise to define name and address structures for a particular locale, but on the international scale it is much harder due to cultural and lingual differences. Previous versions of xNAL defined the name and address structures to a great level of details providing very hierarchical XML structures to express names and addresses in a consistent way.

However, the previous versions were:

- ambiguous in providing multiple options for representing the same information
- complex model for simple representation of name and address data
- difficult to implement as an object model
- too complex for many applications

In many cases xNAL family was used as a basis for a localized standards that were much simpler, but not truly interoperable on a global scale. The derived standards were mainly about scaling it down to a simpler and lighter version that would meet the local requirements.

CIQ TC recognized the need for simplifying the specifications while keeping them locale independent and interoperable on a global scale.

2.2 What is in scope in this version

- Ensure all the overall expressive power of version 2.0 is not lost
- The specification will include W3C XML schemas
- Name and address examples defined using version 2.0 will be represented in version 3.0
- High level UML models of the schemas

2.3 What is out of scope in this version

- DTDs
- Privacy and security issues connected to exchanging and storing personal information
- Data exchange methods and procedures for party information
- Messaging protocol for exchange of party information
- Validation/verification of party information
- Formatting, labeling, or sorting of party information
- API specifications
- Backward compatibility with previous versions

3 CIQ TC Specifications Version 3.0

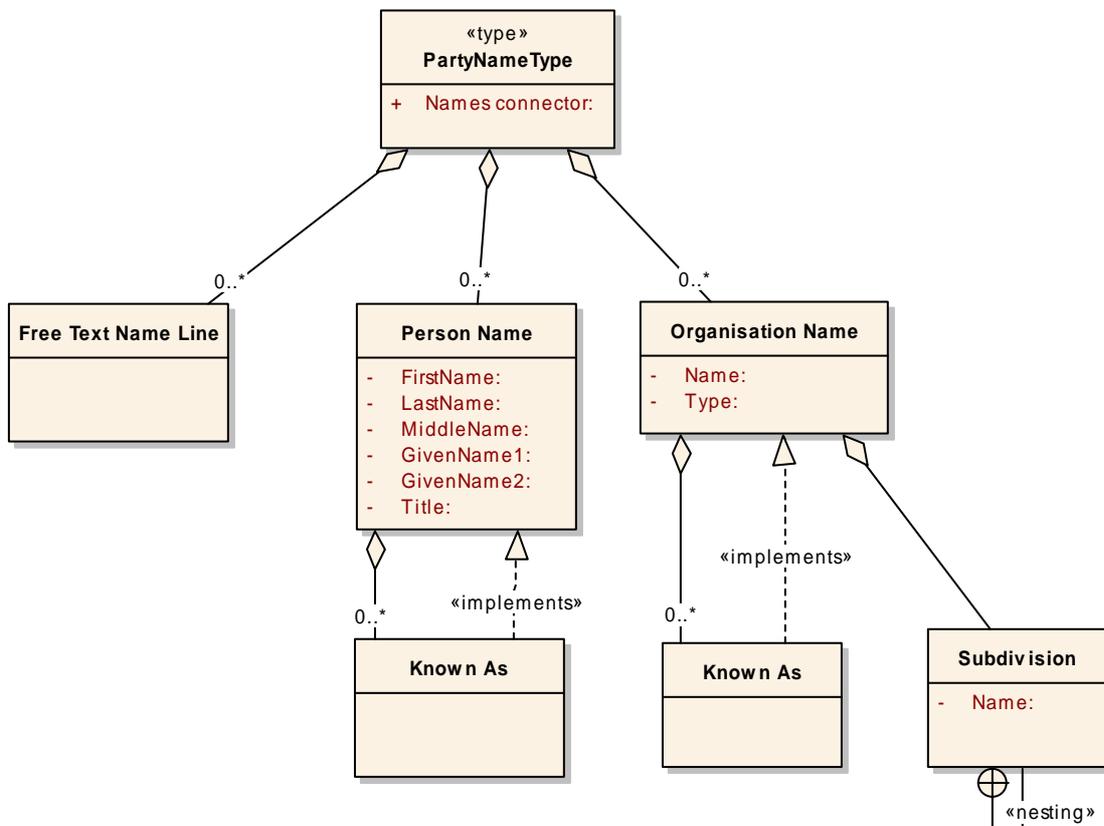
This section provides a brief overview of the CIQ TC specifications (Version 3.0).

3.1 Extensible Name Language (xNL)

xNL defines an XML structure to represent party name data. An example of Party is “customer”. A party could be a “Person” or an “Organization”. An “Organization” could be educational institutions like school, university, college, etc, clubs, associations, industry groups, not-for-profit bodies, consortiums, etc.

xNL was designed to handle international name data that are culturally and geographically specific. For example, the concept of given name and family names do not exist in some cultures, e.g. in some regions of India.

xNL can handle names in over 36 formats and it is extendable. The diagram below illustrates a high level UML model of xNL.



Example – simple person name

```
<n:PartyName>
  <n:PersonName>
    <n:NameLine>Mr Jeremy Apatuta Johnson</n:NameLine>
  </n:PersonName>
</n:PartyName>
```

Example – complex person name

```

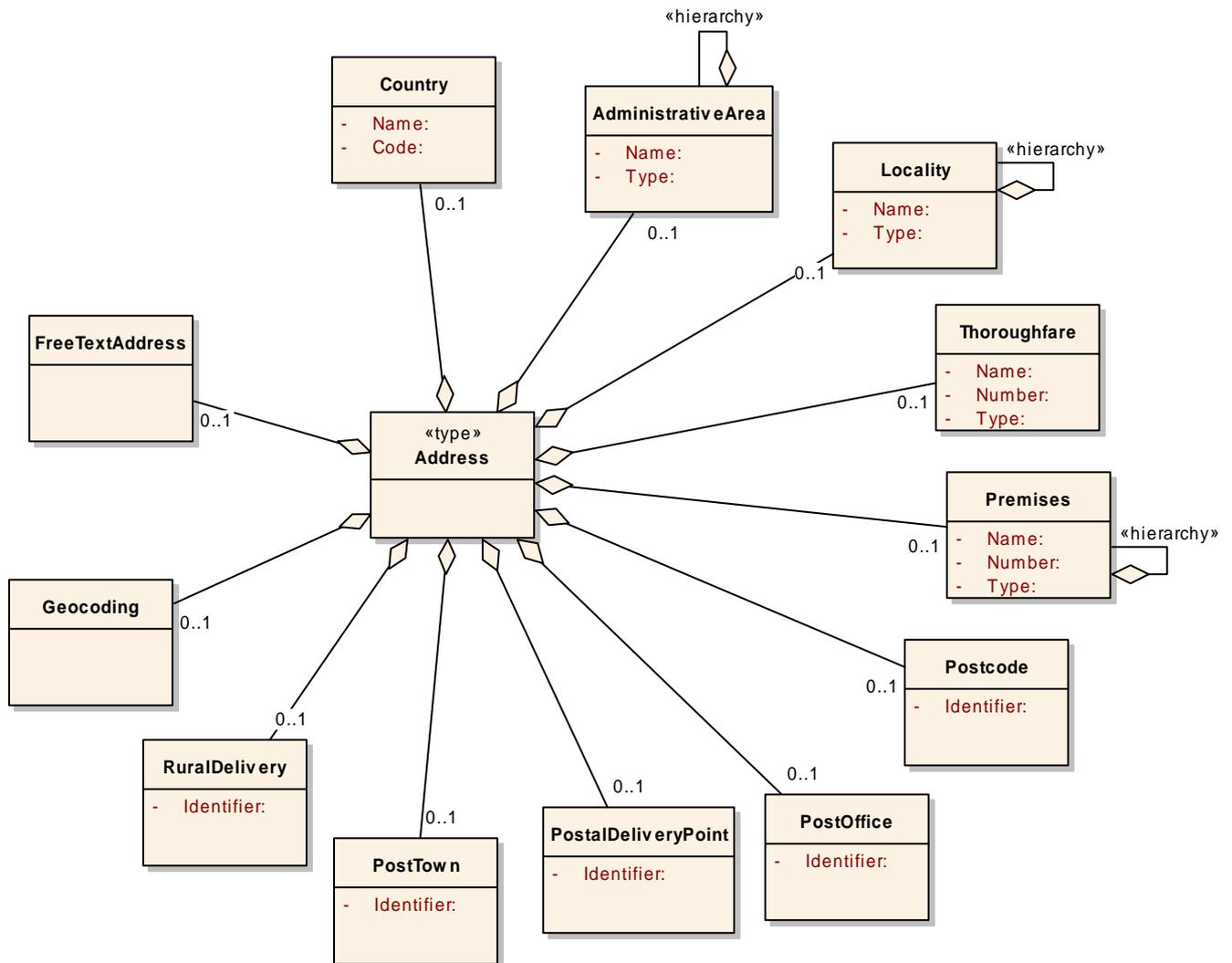
<n:PartyName>
  <n:PersonName>
    <n:NameElement Abbreviation="true" ElementType="Title">Mr</n:NameElement>
    <n:NameElement ElementType="FirstName">Jeremy</n:NameElement>
    <n:NameElement ElementType="MiddleName">Apatuta</n:NameElement>
    <n:NameElement ElementType="LastName">Johnson</n:NameElement>
    <n:NameElement ElementType="GenerationIdentifier">III</n:NameElement>
    <n:NameElement ElementType="Title">PhD</n:NameElement>
  </n:PersonName>
</n:PartyName>
    
```

3.2 Extensible Address Language (xAL)

xAL defines an XML structure to represent address data. An address could include but not limited to any of the following types:

Airport, Business/Commercial Parks, Caravan Parks, Community Developments, Dual (Primary and Secondary), Educational institutions, Entertainment/Recreation Parks, Hospitals, Large Mail Users, Marinas, Military, Ports, Retirement Villages, Resorts, Royal Highness, Rural (with land, air and water access), Sporting Venues, Territories, Tribal, Simple Urban, Complex Urban, Utility Urban, Ranged Urban, Villages, Canals, Banks, etc

xAL can handle addresses of 245+ countries in over 130 formats. The diagram below illustrates a high level UML model of xAL.



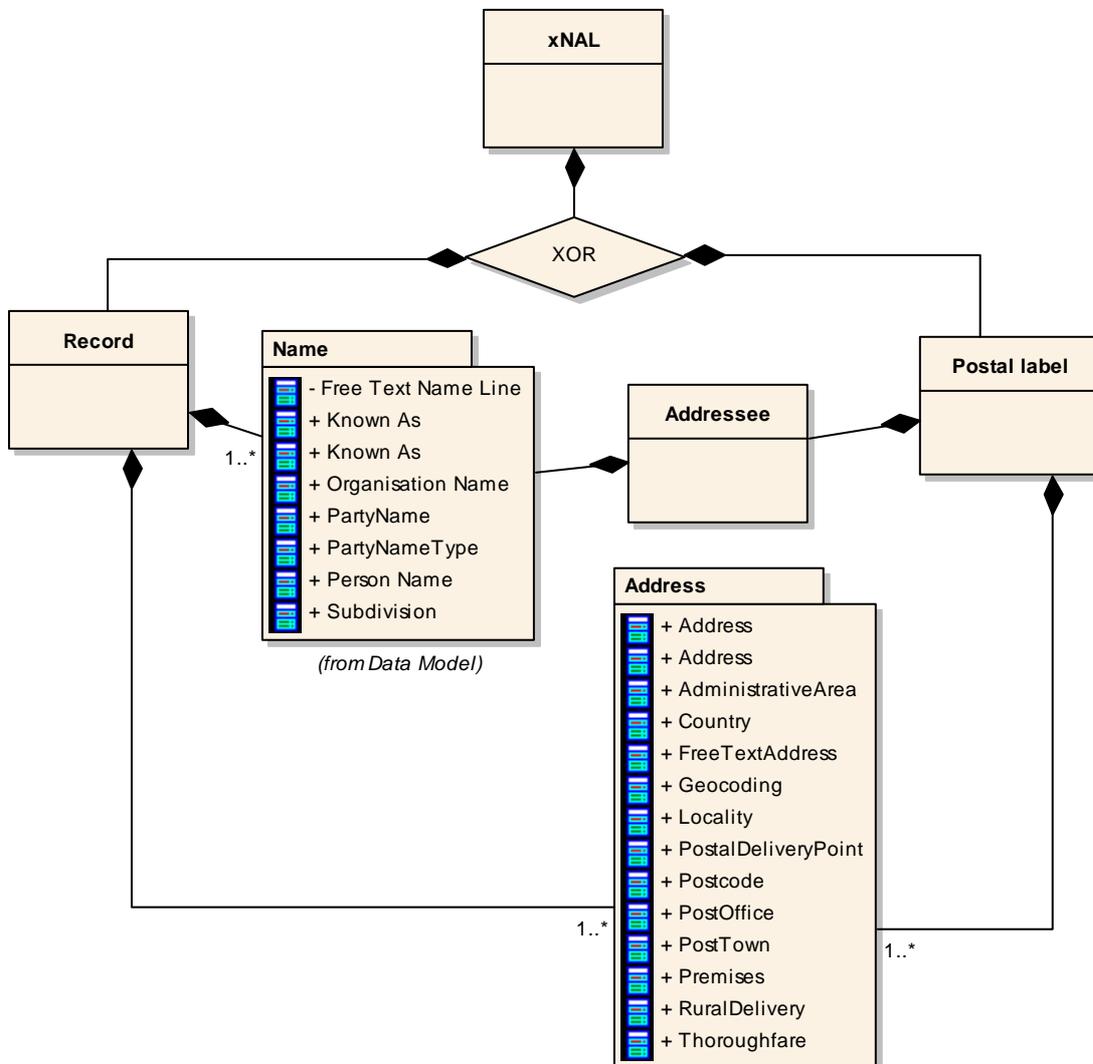
Example – simple address

```

<a:Address>
  <a:AdministrativeArea>
    <a:Name>WA</a:Name>
  </a:AdministrativeArea>
  <a:Locality>
    <a:Name>OCEAN REEF</a:Name>
  </a:Locality>
  <a:Thoroughfare>
    <a:NameElement>16 Patterson Street</a:NameElement>
  </a:Thoroughfare>
</a:Address>
    
```

3.3 Extensible Name and Address Language (xNAL) Version 3.0

xNAL defines an XML structure to represent name and address data bound together. xNAL utilizes XML structures from xNL and xAL specifications. The diagram below illustrates a high level UML model of xNAL version 3.0.



Technical Overview

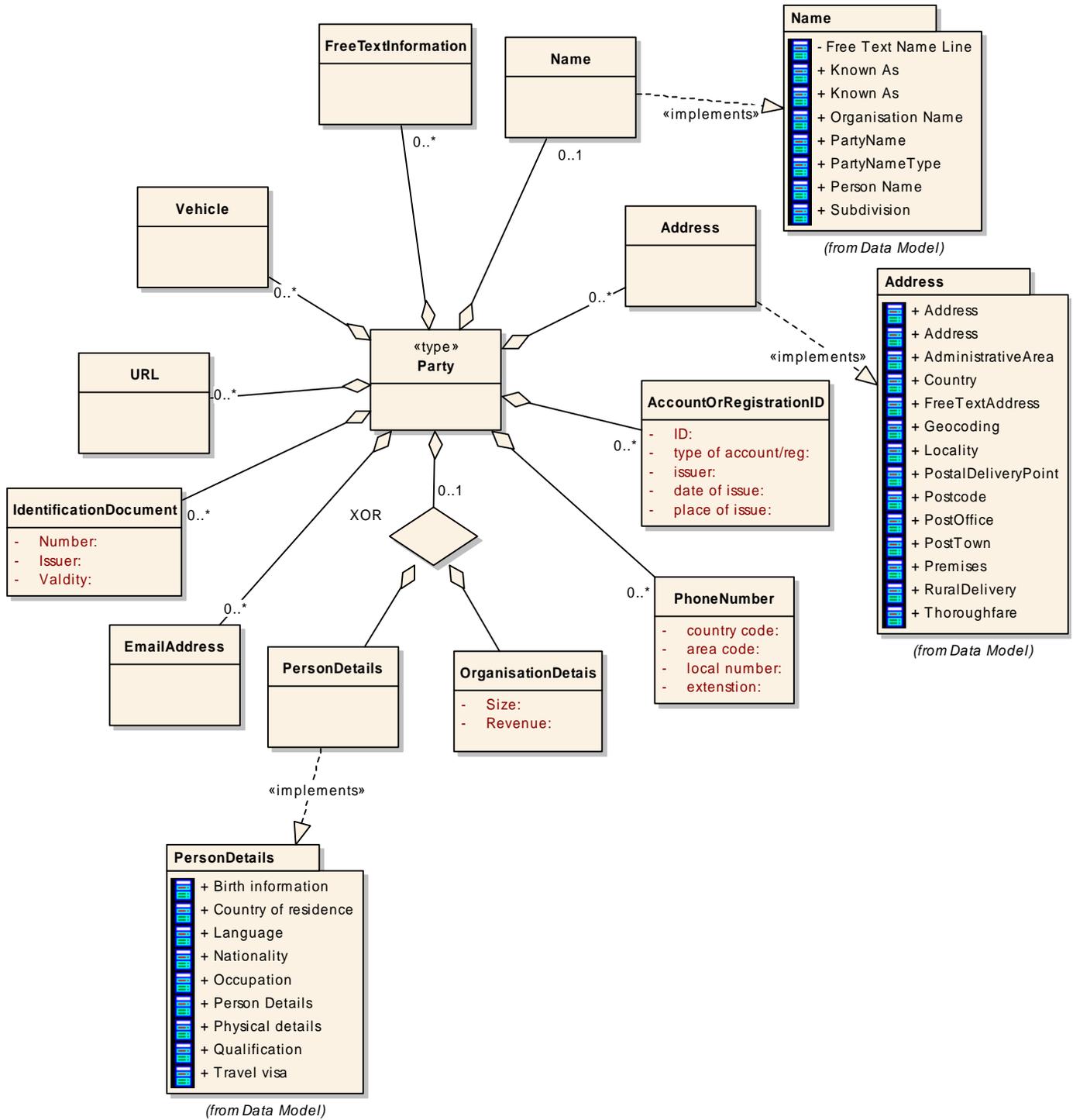
Example

Mr H G Guy, 9 Uxbridge Street, Redwood, Christchurch

```
<xnal:Record>
  <n:PartyName>
    <n:NameLine>Mr H G Guy</n:NameLine>
  </n:PartyName>
  <a:Address>
    <a:Locality>
      <a:Name>Christchurch</a:Name>
      <a:SubLocality>
        <a:Name>Redwood</a:Name>
      </a:SubLocality>
    </a:Locality>
    <a:Thoroughfare>
      <a:Number>9</a:Number>
      <a:NameElement>Uxbridge Street</a:NameElement>
    </a:Thoroughfare>
  </a:Address>
</xnal:Record>
```

3.4 Extensible Party Information Language (xPIL)

xPIL defines an XML structure to represent party-centric data. Party-centric data includes name, address, e-mail address, telephone numbers, identification details (e.g. passport, license number, identification card, etc), vehicle details, account details, etc. The diagram below illustrates a high-level UML view of xPIL version 3.0



3.5 Extensible Party Relationships Language (xPRL)

xPRL defines a consistent way of using xLink to represent party relationships. Party relationships could be:

- Person to Person relationships
- Person to Organization relationships, and
- Organization to Organization relationships

Release date for version 3.0 of this specification not set yet.

4 Practical applications of CIQ TC specifications

Some readers may find it hard to get to grips with the CIQ TC specifications family. This section is an informative guide to help you get started.

4.1 Where to start

Consider doing the following:

- Clearly define your requirements and goals of using CIQ Specifications
- Complete reading this document (15 minutes)
- Study the XML examples of the schemas (30 minutes). Examples are provided in the same download as the schemas.
- Study the schema diagrams (15 minutes). You can browse the schemas using an XML editor or use HTML documentation provided as part of every CIQ TC specification
- Try to build the structures you need using the schemas and your sample data (20 minutes). You may want to use an XML editor that provides information from schema `xs:annotation` elements to help you understand the meaning of the elements and attributes.

4.2 Don't get confused – keep it simple

Choose the simplest out of two options. xNL, xAL and other CIQ TC specs allow for more than one way to represent information. You need to choose the one that suits your particular task. Version 3.0 allows you to customize the specifications to meet your requirements without affecting the structure of the schemas. However, please ensure that what you have customized is agreeable with other party that exchanges data with you (e.g. applications, end users, external parties) to achieve interoperability.

4.3 Data exchange

CIQ TC specifications can be used to organize data exchange of party information or just names and addresses. It is likely that just CIQ TC specs are not enough to organize such an exchange as it requires some messaging mechanisms and additional information such as metadata.

CIQ TC recommends that reusable elements from the CIQ TC schemas are used inside other namespaces or wrappers. This will ensure that the original namespaces remain intact while additional information is still provided.

4.4 Output formatting

CIQ TC specifications do not have any means to specify the formatting of the data. It is up to the application to decide which formatting suits best. It is recommended to preserve the original order of elements to assist with correct output formatting. Remember, that addresses, for example, may begin with the finest details (e.g. flat number) in some locales or with country name in the other. Preserving the original order is important.

4.5 Schema extensions

It is possible to extend CIQ TC schemas within some allocated boundaries to meet specific application or locale requirements. The extensions can be of two types:

- Any element can have any number of attributes from the non-target namespace, which means you can include some other attributes not specified by the schema.
- Enumerations can be changed and they were intentionally placed in a separate “include” file.

Adding new elements to the schema is not permitted – use wrappers instead.

4.6 Data mapping challenges

The main challenge in standardising name and address and even party data structures is in a potentially infinite number of ways they can be presented for different applications, different cultures and locales.

4.6.1 Application diversity

For example a simple e-commerce database may have name as one field, address as a free-text 3-field set and other party information in a dozen of other fields. It may be sufficient for that particular data usage scenario.

A larger bank may be interested in a more detailed name and address structure to allow business intelligence applications to do their analysis.

The differences in complexity between these two examples present a great challenge finding a common form of representing the data so that it is attractive to all potential participants.

4.6.2 Cultural diversity

Name and address presentation formats vary between cultures elevating the importance of breaking down the structure and preserving the original meaning of the elements so that the name or address can be correctly restored at a later time. It is virtually impossible to fit all these diverse views into a single name and address specification that that is also specific to a particular culture. Some balanced approach is required to meet the semantic and presentation variations and requirements in one specification. It is the goal of CIQ TC to achieve such a balance.

India is a good example of cultural diversity with people from different ethnic backgrounds, languages (officially 14 national languages) and religions. In some places there is no concept of family name or given name or surname or first name or middle name or last name. They have the following name types that can be used as part of a person's name:

Grand father name, Great grand father name, Father's name, Mother's name, Native Place name, Tribal name, Caste name, Husband's name, Birth name, etc.

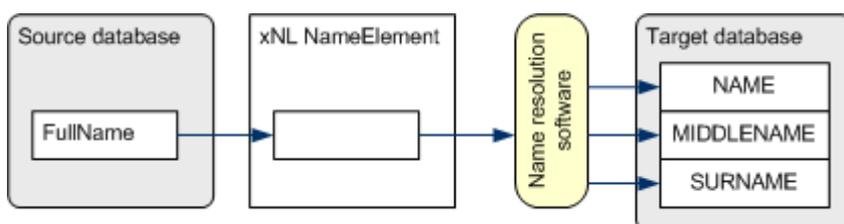
Addresses are culture and locale specific too. There is usually a great degree of freedom how one can write an address with any information that is specific to the geographic location/locale and it still reaches the destination. For example, in countries like Thailand, addresses include the names of the banks, or canals instead of streets. The concept of neither the postal code nor the locality applies to some countries. In some countries an address is attached to the number of a postal van that delivers the mail to the destination as the van is responsible for delivering mails to a certain area/streets in an area.

4.6.3 CIQ TC solution

CIQ TC provides a solution that can take and persist with the information in the form it was originally provided without any loss of semantics so that the information can be mapped to some target structure with a minimal effort.

However, CIQ TC does not provide a solution for mapping a simple source structure to a complex target one as it would require parsing and "understanding" the information carried in the structure itself. Any solution to this problem is out of scope for CIQ TC.

The diagram below shows how a simple one-field data model can be mapped to another complex data model through xNL, but with help of "name resolution software" to separate a *full name* into *name*, *middle name* and *surname*:



Appendix A. Notices

Copyright © OASIS Open 2005. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.