# Customer Information Quality Party Relationships (xPRL) Specification Version 3.0

## Committee Specification 01

## 11 November 2009

**Specification URIs:**
**This Version:**
http://docs.oasis-open.org/ciq/v3.0/xPRL/cs01/specs/ciq-xprl-specs-cs01.html
http://docs.oasis-open.org/ciq/v3.0/xPRL/cs01/specs/ciq-xprl-specs-cs01.doc (Authoritative)
http://docs.oasis-open.org/ciq/v3.0/xPRL/cs01/specs/ciq-xprl-specs-cs01.pdf

**Previous Version:**
http://docs.oasis-open.org/ciq/v3.0/xPRL/prd01/specs/ciq-xprl-specs-prd01.html
http://docs.oasis-open.org/ciq/v3.0/xPRL/prd01/specs/ciq-xprl-specs-prd01.doc
http://docs.oasis-open.org/ciq/v3.0/xPRL/prd01/specs/ciq-xprl-specs-prd01.pdf

**Latest Version:**
http://docs.oasis-open.org/ciq/v3.0/xPRL/specs/ciq-xprl-specs.html
http://docs.oasis-open.org/ciq/v3.0/xPRL/specs/ciq-xprl-specs.doc
http://docs.oasis-open.org/ciq/v3.0/xPRL/specs/ciq-xprl-specs.pdf

**Technical Committee:**
OASIS Customer Information Quality TC

**Chair:**
Ram Kumar (kumar.sydney@gmail.com)

**Editor:**
Ram Kumar (kumar.sydney@gmail.com)

**Related work:**
This specification replaces or supercedes:

- OASIS CIQ extensible Customer Relationships Language (xCRL) V2.0 Committee Specification

**Declared XML Namespace(s):**
urn:oasis:names:tc:ciq:xprl:3

**Abstract:**

This Technical Specification defines the extensible Party Relationships Language (xPRL) specifications of OASIS Customer Information Quality Specifications Version 3.0.

**Status:**
This document was last revised or approved by the OASIS CIQ Technical Committee (TC) on the above date. The level of approval is also listed above. Check the current location noted above for possible later revisions of this document. This document is updated periodically on no particular schedule.

Technical Committee members should send comments on this specification to the Technical Committee's email list. Others should send comments to the Technical Committee by using the

"Send A Comment" button on the Technical Committee's web page at http://www.oasis-open.org/committees/ciq/.

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 Technical Committee web page (www.oasis-open.org/committees/ciq/ipr.php.

The non-normative errata page for this specification is located at http://www.oasis-open.org/committees/ciq/.

# Notices



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.

OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification.

OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this specification by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification. OASIS may include such claims on its website, but disclaims any obligation to do so.

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Committee Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The names "OASIS", “CIQ”, “xNL”, “xAL”, xNAL”, “xPIL”, “xPRL”, “xCIL”, “xCRL” , “Genericode”, and “UBL” are trademarks of OASIS, the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see http://www.oasis-open.org/who/trademark.php for above guidance.

# Table of Contents

# 1 Normative References

Following are the documents that users of this specification SHOULD read and understand:

- OASIS Customer Information Quality Specifications V3.0 – Name, Address and Party, Committee Specification 02, March 2008, http://www.oasis-open.org/committees/ciq
- OASIS Codelist Representation (Genericode) Version 1.0, Committee Specification 01, December 2007, http://www.oasis-open.org/committees/codelist
- Context Value Association, Working Draft 0.4, April 2008, http://www.oasis-open.org/committees/codelist
- OASIS Code List Adaptation Case Study (OASIS CIQ), 2007, http://www.oasis-open.org/committees/codelist

# 2 Name, Address, Party, and Party Relationship

## 2.1 Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in [RFC2119].

While RFC2119 permits the use of synonyms, to achieve consistency across specifications, “MUST” is used instead of “SHALL” and “REQUIRED”, “MUST NOT” instead of “SHALL NOT”, and “SHOULD” instead of “RECOMMENDED” in this specification. To enable easy identification of the keywords, uppercase is used for keywords.

## 2.2 Definitions

Following are the core entities and its definitions used by CIQ TC:

**Name**

Name of a person or an organisation

**Address**

A physical location or a mail delivery point

**Party**

A Party COULD be of two types namely,

- Person
- Organisation

An Organisation COULD be a company, association, club, not-for-profit, private firm, public firm, consortium, university, school, etc.

Party data consists of many attributes (e.g. Name, Address, email address, telephone, etc) that are unique to a party. However, a person or organisation’s name and address are generally the key identifiers (but not necessarily the unique identifiers) of a “Party”. A “Customer” is of type “Party”.

**Party Relationship**

Pairwise affiliation or association between two people, between two organisations, or between an organisation and a person.

xPRL supports chains of interlocking pairwise party relationships, linked by common members.

# 3 Extensible Party Relationships Language (xPRL) Version 3.0

## 3.1 Pre-requisite

It is a pre-requisite that users MUST study the "**OASIS CIQ V3.0 Name, Address and Party Committee Specification 02**" that was released in November 2008 before reading this document. The specification is located in: http://www.oasis-open.org/committees/ciq.

This Party Relationships specification uses the same design concepts and other industry specifications used by OASIS CIQ V3.0 CS02 Name, Address, and Party specifications.

When OASIS CIQ Name, Address and Party Version 3.0 Committee Specifications were originally released in November 2007, xPRL version 3.0 was not part of the release. However, the following documents released in November 2007 and subsequently released as Committee Specification 02 in November 2008 are applicable to this specification also and hence, SHOULD be read in conjunction with this specification.

- CIQ Committee Specifications Version 3.0 CS02 - Name, Address and Party
- CIQ Specifications Version 3.0 – General Introduction and Overview
- CIQ Specifications Version 3.0 – Release Notes
- CIQ Specifications Version 3.0 – Technical Overview
- CIQ Specifications Version 3.0 – Frequently Asked Questions
- CIQ Specifications Version 3.0 CS02 – Package Overview

To extract and install the xPRL related schema, documents and examples, read the "CIQ Specifications Version 3.0 - xPRL Package Overview" document located in the downloaded package's directory "\ciq-xprl-v3\ciq\v3.0\supp".

## 3.2 The need for a Party Relationships Standard

The rapid adoption of e-business has created a new world of interoperability between organisations, systems, processes, platforms, tools and, most importantly, data. When we start to consider party management initiatives (e.g. CRM/eCRM, Single/360 degree View of a Party, Customer Information Warehouse, Customer Data Management, Party Data Management, Master Data Management), there are many other factors than software license fees and customisation, training, maintenance that raise the cost of deployment. Integration of systems, for example, can be a far more significant and costly challenge. That is because, in most large enterprises, party information is captured and stored in multiple "proprietary" systems. Bringing it all together for analysis in a party information management system usually involves time-consuming integration using the proprietary APIs provided by CRM and other enterprise software vendors. Backend systems integration is where most of the real cost – and risk – of implementing CRM and ERP systems lies. Many of these implementations have significantly under delivered because cost has prohibited them from interfacing with other key systems.

If there is a standard way of defining party information and relationships between parties that is vendor neutral and open (i.e., independent of tools, systems, languages and platforms) and enabled portability and interoperability of data, then it would be possible to reduce the expensive and complex Integration problems associated with new business initiatives.

extensible Party Relationships Language (xPRL) specification is intended to meet this requirement. xPRL, is a set of XML vocabulary specifications for defining party (person or organisation) characteristics such as name, address, age, party identifier, e-mail address and so on that will assist in uniquely identifying a party. In addition, xPRL describes, in a standard way, relationship(s) between parties. As currently

defined, xPRL enables users to describe relationships such as person-to-person, person-to-organisation or organisation-to-organisation in a standard way. So, if a CRM system and, say, an Enterprise Resource Planning system both understood xPRL definitions via its interfaces or through a middleware, they could interoperate without needing expensive, custom integration. This would accelerate the time taken to deploy such systems and allow them to interact more readily with a wider range of other systems.

There are no standards for representing party relationships in industry and xPRL helps fill this gap by defining the nature of relationship between two or more parties and detailed personal profile of each party involved in the relationship. For detailed personal profile of each party (e.g. name, address, contact details, party characteristics), xPRL uses OASIS xPIL v3.0 Specification.

## 3.3 xPRL v3.0 Schema Files

Following are the different schemas produced for xPRL version 3.0:

| Schema File name | Description | Comments |
|---|---|---|
| xPRL.xsd **(formerly known as "xCRL.xsd")** | Entity Party Relationship | Defines a set of reusable types and elements for relationships between parties |
| xPRL-types.xsd | Entity Party Relationship Enumerations | Defines a set of enumerations to support Relationship entity |
| *.gc files | Entity Party Relationship | Defines a set of enumerations/code lists in genericode |

*xPRL.xsd* reuses the OASIS CIQ V3.0 XML schemas of Name, Address and Party entities.

## 3.4 Namespaces Used

Following are the namespaces used in the specification:

| Entity | Namespace | Recommended Prefix | Schema Files |
|---|---|---|---|
| Party Relationship | urn:oasis:names:tc:ciq:xprl:3 | xprl (or) r | xPRL.xsd xPRL-types.xsd |
| xLink | http://www.w3.org/1999/xlink | xlink | xlink-2003-12-31.xsd |

## 3.5 Out of Scope

This specification does not cover the areas that are considered out of scope by CIQ Specifications V3.0 as defined in the following document:

- Customer Information Quality Specifications version 3.0 CS02, General Introduction and Overview, September 2008

In addition to the above, this specification does not cover the following as these are outside the scope of the CIQ technical committee:

- Relationship description about party related "non personal profile entities" such as financial/business transactions, information, product information, service information, etc
- Privacy and access policies, access logging, tracking, and control of party data and between parties

# 4 Types of Party Relationships Supported

Following are the core types of party relationships and the contextual role each party plays in the relationship that are supported by this specification. A party could be an individual (person or an organisation), or a group of persons or organisations.

## 4.1 Person(s) To Person(s) Relationship(s)

Some examples of Person(s) to Person(s) relationship(s) are:

- Mrs. Mary Johnson and Mr. Patrick Johnson, where Mary is the *"Wife"* of Patrick and Patrick is the *"Husband"* of Mary
- Mrs. Mary Johnson and Mr. Patrick Johnson *"IN TRUST FOR"* Mr. Nick Johnson, where Mary and Patrick are the *"Trustees"* of Nick and Nick is the "Beneficiary"
- Mrs. Mary Johnson, Care of Mr. Patrick Johnson, where Mary is *"Dependent"* on Patrick
- Personal/Business contacts
- Group of people have a relationship with another group of people. E.g. Family to Family relationship
- Family tree and profiles of each individual person in the tree

## 4.2 Person(s) To Organisation(s) Relationship(s), and vice versa

Some examples of Person(s) to Organisation(s) relationship(s) are:

- Mrs. Mary Johnson and Mr. Patrick Johnson *"DOING BUSINESS AS"* Johnson & Associates, where Mary and Patrick are persons who are jointly doing a business under the name of a trading entity called "Johnson & Associates"
- Mr. Ram Kumar, Care of Digeridoo Pty. Ltd, where Ram is the person and Digeridoo Pty. Ltd. is the Company
- Mrs. Mary Johnson and Mr. Patrick Johnson *"IN TRUST FOR"* Mr. James Johnson "DOING BUSINESS AS" Johnson and Associates
- Mr. Ram Kumar is the "Chief Technical Officer" of XYZ Company, where Ram Kumar has a designation of Chief Technical Officer and is an employee of XYZ Company
- Ram Kumar's business (organisation) contacts
- Ram Kumar of XYZ Company is a consultant/contractor/supplier to ABC Company, where Ram Kumar is an employer of XYZ Company and XYZ Company's client is ABC Company
- Ram Kumar is an employee of UVR Company
- Organisation and its employees (e.g. Organisation structure)

## 4.3 Organisation(s) To Organisation(s) Relationship(s)

Some examples of Organisation(s) to Organisation(s) relationship(s) are as follows:

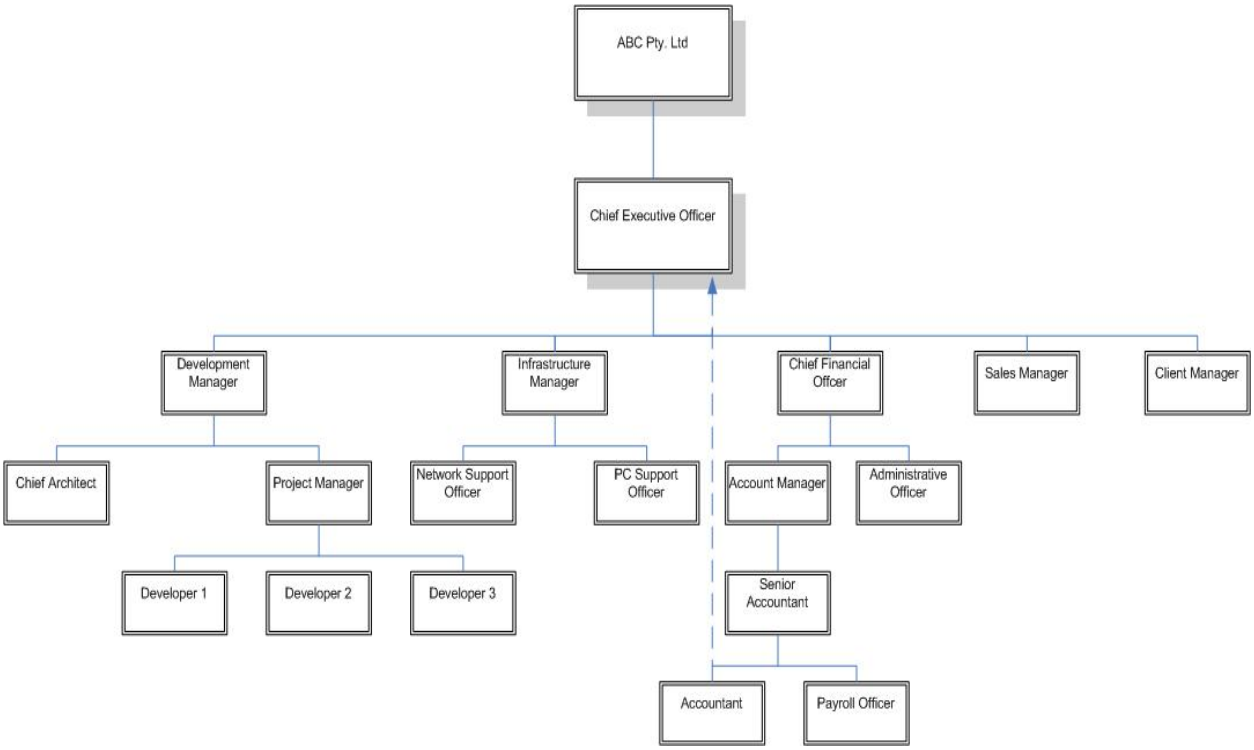
- Company A is a *subsidiary* of Company B
- Company A is the *parent* of Company B
- Company A, Company B and Company C are the subsidiary companies of Company D
- Richardson and Wrench *"TRADING AS"* Johnson Associates, Inc
- Richardson and Wrench is a *"LAND LINE CUSTOMER OF"* AT&T and is also a *"SUPPLIER"* to AT&T
- Company A's business partners are Company B, Company C, and Company D .

- Group (not necessarily a legal entity) of companies have a relationship with another group (not necessarily a legal entity) of companies in bidding a tender
- Golf Club of Turramurra suburb is a *member* of the NSW State Golf Club Association

## 4.4 Complex Party Relationships

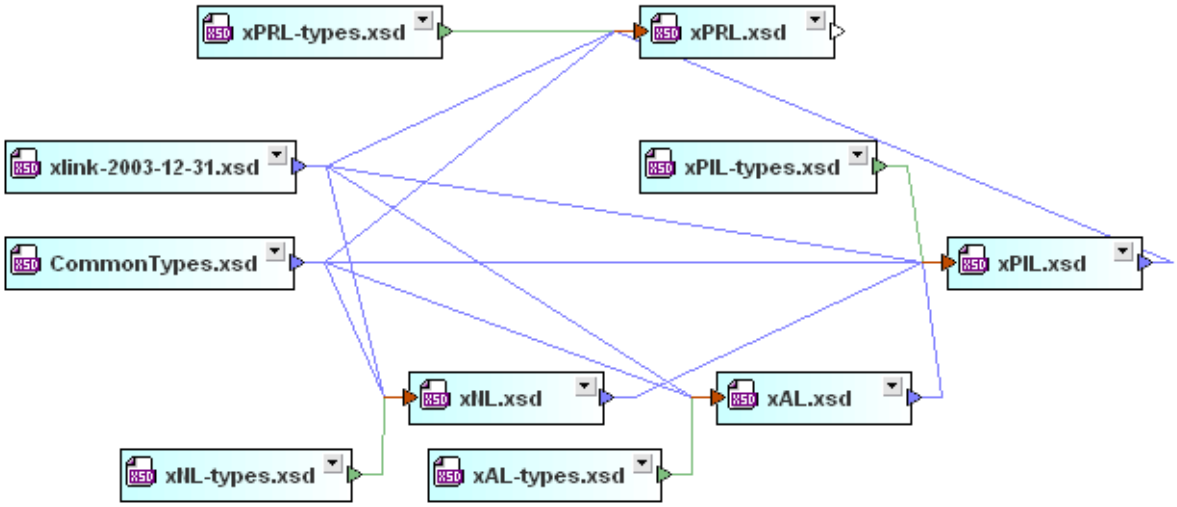
xPRL also provides the capability to define and represent complex relationships that may be hierarchical or deeply nested structure in nature. Examples include:

- Mrs Mary Jackson AND Mr. James Jackson *"IN TRUST FOR"* Mr. Patrick Jackson *"DOING BUSINESS AS"* Jackson and Associates Pty. Ltd *"TRADING AS"* Jackson International Pty. Ltd
- An organisation structure. An example of an organisation structure that can be represented using *xPRL* is shown below.



# 5 xPRL Data Model

*xPRL* links two parties through a "Relationship" entity. The two party entities in the relationship reuse Party entity defined in *xPIL* specification. *xPIL* specification reuses *xNL* and *xAL* specifications. This is shown in the following figure:




At least two parties are required to define a relationship. We classify the two parties as "Party in context/discussion" and "Other Party" For example, if Party "A" has a relationship with Party "B", then Party "A" is "the party in context/discussion" and Party "B" is "the other party", and vice versa. If Party "B" in turn has a relationship with Party "C", then Party "B" is "the party in context/discussion" and Party "C" is "the other party". "The party in context/discussion" does not mean that it has more authority or priority over "the other party". It is just a way of differentiating between two parties that MAY or MAY NOT play equally important roles in the relationship. Given that "Party A" is the subject of discussion, "Party A" is defined as "the party in context/discussion". The following section provides some examples that explain this in detail.

## 5.1 Examples

### 5.1.1 Example 1

```
Mrs. Mary Jackson is the wife of Mr. Patrick Jackson
```

In the above example, if we use Mary Jackson as the "Party" under discussion whose profile and relationships are defined using xPRL, then Mary Jackson is defined as "the party incontext/discussion" and Patrick Jackson is defined as "the other party". Both the parties here play an equally important role in their relationship to each other namely, Mary being the "wife of" Patrick and Patrick being the "husband of" Mary.

### 5.1.2 Example 2

```
Mrs. Mary Jackson is the wife of Mr. Patrick Jackson and Mr. John Jackson is
the brother of Mr.Patrick Jackson
```

In the above example, if Mary Jackson is the "Party" under discussion, then Mary is "the party in context/discussion" and Patrick is "the other party". If this relationship should also include John, then Patrick is now represented as "the party in context/discussion" and John is represented as "the other

party" under xPRL. There is no direct relationship between Mary and John represented here as shown below unless the example explicitly states that "Mary is the sister in law of John":

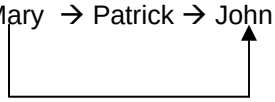
Mary Jackson -> Patrick Jackson -> John Jackson

### 5.1.3 Example 3

```
Mrs. Mary Jackson is the wife of Mr. Patrick Jackson. Mr. John Jackson is the
brother of Mr.Patrick Jackson and is the brother-in-law of Mrs. Mary Jackson
```

In the above example, if Mary Jackson is the "Party" under discussion, then Mary is "the party in context/discussion" and Patrick is "the other party". Mary also has a relationship with John. Therefore, John is also defined as "the other party". To define the relationship between Patrick and John, Patrick is defined as "the party in context/discussion" and John is defined as "the other party" as shown below:

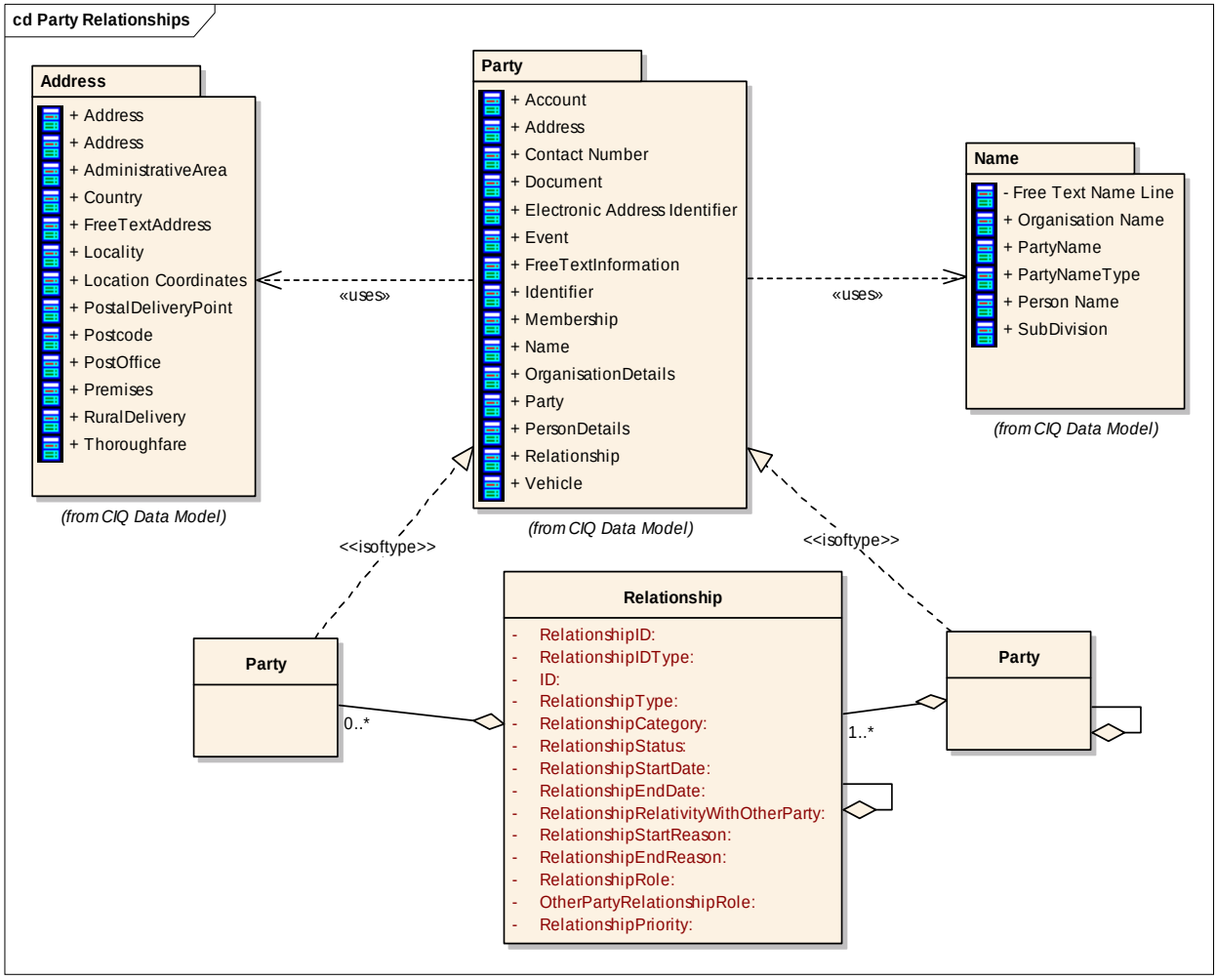
Mary → Patrick → John



The data model of *xPRL* specification is shown in the next section.

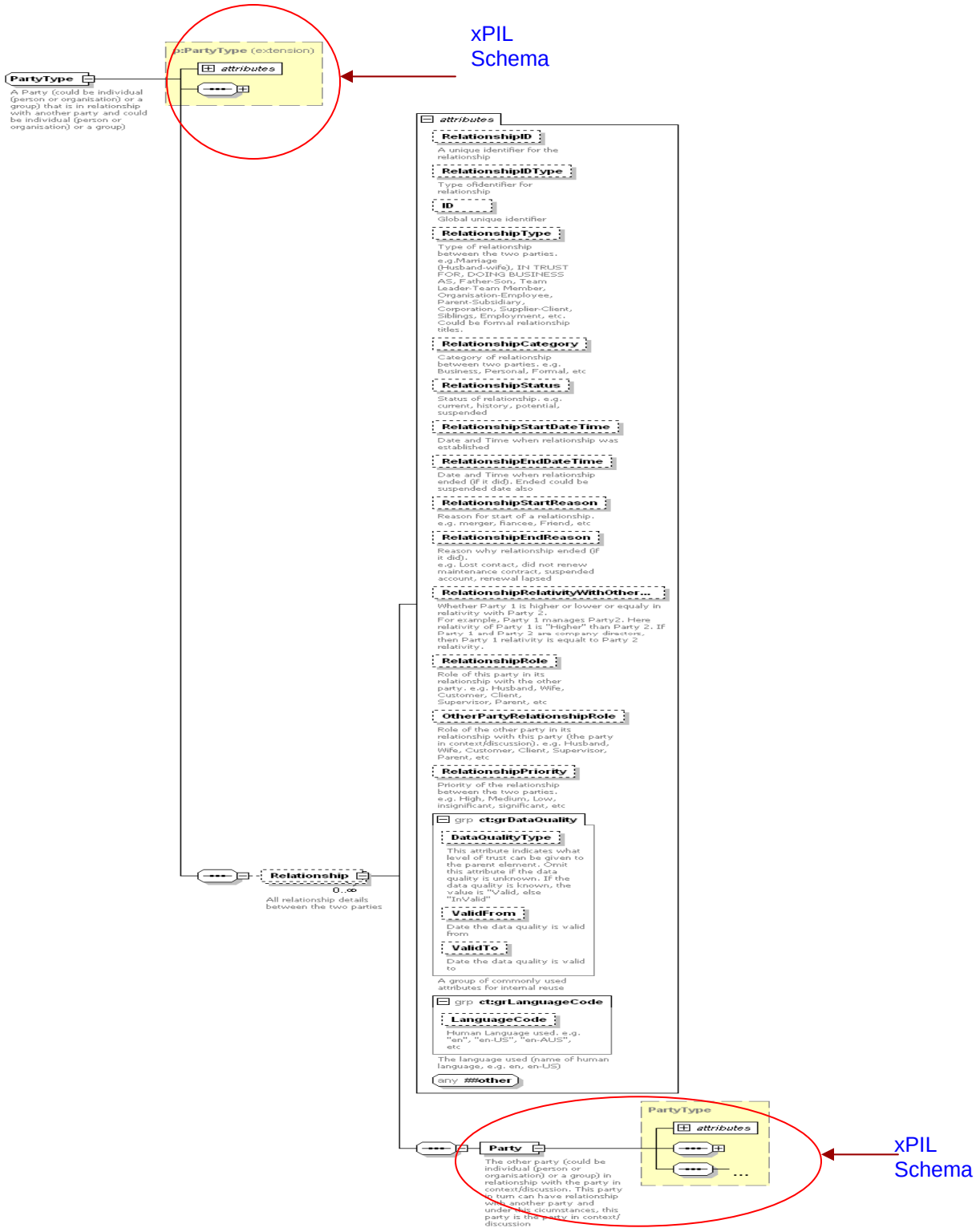
## 5.2 xPRL Entity-Relationship Model

The diagram below shows the Entity Relationship model of *xPRL* specification.



## 5.3 xPRL XML Schema Model

The figure below shows the XML Schema model of *xPRL* specification.




The entity “Relationship” in the schema has about 11 attributes that defines the relationship attributes such as type, status, start and end dates.

# 6 Entity “Party Relationships”

## 6.1 Data Types

All elements and attributes in *xPRL* schema have strong XML data types.

All free-text values of elements (text nodes) and attributes are constrained by XML simple type “*normalizedString*” (collapsed white spaces) defined in *CommonTypes.xsd*. Other XML data types are also used throughout the schema.

## 6.2 Code Lists (Enumerations)

Use of code lists/enumerations is identical to use of code lists for entity “*Name*”, “*Address*”, and “*Party*” specifications. This is explained in the OASIS CIQ V3.0 Name, Address and Party specification document (see section 3.4).

Code lists/enumerations used in *xPRL* for code list option 1 reside in an “include” *xPRL-types.xsd*. Code lists/enumerations used in *xPRL* for code list option 2 reside as “.gc” genericode files.

**NOTE:** The code list/enumeration values for different enumeration lists that are provided as part of the specification are not complete. They only provides some sample values (and in most cases no values) and it is up to the end users to customise them to meet their data exchange requirements if the default values are incomplete, not appropriate or over kill

## 6.3 Order of Elements and Presentation

Order of name elements MUST be preserved for correct presentation. This is explained in the OASIS CIQ V3.0 Name, Address and Party specification document (see section 3.6).

## 6.4 Data Mapping

Mapping data between *xPRL* schema and a database is similar to that of entity “*Name*”, “*Address*”, and “*Party*” as described in the OASIS CIQ V3.0 Name, Address and Party specification document (see section 3.7).

## 6.5 Data Quality

*xPRL* schema allows for data quality information to be provided as part of the entity using attribute *DataQuality*. This is explained in the OASIS CIQ V3.0 Name, Address and Party specification document (see section 3.8).

## 6.6 Extensibility

All elements in *Party* namespaces are extensible as described in the OASIS CIQ V3.0 Name, Address and Party specification document (see section 3.9) are applicable to this specification too.

## 6.7 Linking and Referencing

All linking and referencing rules described in the OASIS CIQ V3.0 Name, Address and Party specification document (see section 3.10) are applicable to this specification too.

## 6.8 ID Attribute

Use of attribute ID is described in the OASIS CIQ V3.0 Name, Address and Party specification document (see section 3.11) are applicable to this specification too.

## 6.9 Schema Conformance

Any XML documents produced MUST conform to the CIQ Specifications Schemas namely, *xPRL.xsd, xNL.xsd, xAL.xsd, xNAL.xsd* and *xPIL.xsd,* i.e. the documents MUST be successfully validated against the Schemas. This assumes that the base schemas MUST be modified.

If Option 2 for Code List is used, all genericode files MUST conform to the Genericode XML Schema, i.e. all genericode files MUST successfully validate against the schema.

Any customisation of the code list files based on Option 1 MUST be well formed schemas.

## 6.10 Schema Customisation Guidelines

Schema customisation rules and concepts as described in the OASIS CIQ V3.0 Name, Address and Party specification document (see section 3.13) are applicable to this specification too.

## 6.11 xPRL Examples

### 6.11.1 Person To Person Relationship

*Mrs Mary Jackson and Mr. James Jackson, where Mary Jackson is the "wife of" James Jackson*

```
<r:Party>
  <p:PartyName>
    <n:PersonName>
       <n:NameElement>Mrs.Mary Jackson</n:NameElement>
    </n:PersonName>
  </p:PartyName>
   <r:Relationship r:RelationshipType="Marriage"
    r:Party1RelationshipRole="Wife"
    r:Party2RelationshipRole="Husband">
      <r:Party p:PartyType="Person">
        <p:PartyName>
           <n:PersonName>
             <n:NameElement>Mr. James Jackson</n:NameElement>
           </n:PersonName>
        </p:PartyName>
     </r:Party>
   </r:Relationship>
</r:Party>
```

### 6.11.2 Organisation To Organisation Relationship

*ABC Pty. Ltd is a subsidiary of XYZ Pty. Ltd*

```
<r:Party>
  <p:PartyName>
    <n:OrganisationName>
       <n:NameElement>ABC Pty. Ltd</n:NameElement>
    </n:OrganisationName>
  </p:PartyName>
   <r:Relationship r:RelationshipType="Corporation"
    r:RelationshipRole="Subsidiary Company"
    r:OtherPartyRelationshipRole="Parent Company">
      <r:Party>
        <p:PartyName>
           <n:OrganisationName>
             <n:NameElement>XYZ Pty. Ltd</n:NameElement>
           </n:OrganisationName>
        </p:PartyName>
       </r:Party>
   </r:Relationship>
</r:Party>
```

### 6.11.3 Organisation To Person Relationship

*ABC Pty. Ltd is the employer of Ram Kumar*

```
<r:Party>
  <p:PartyName>
    <n:OrganisationName>
       <n:NameElement>ABC Pty. Ltd</n:NameElement>
    </n:OrganisationName>
  </p:PartyName>
   <r:Relationship r:RelationshipType="Employment"
    r:RelationshipRole="Employer"
    r:OtherPartyRelationshipRole="Employee">
      <r:Party>
        <p:PartyName>
           <n:PersonName>
             <n:NameElement>Ram Kumar</n:NameElement>
           </n:PersonName>
        </p:PartyName>
       </r:Party>
   </r:Relationship>
</r:Party>
```

### 6.11.4 Person To Person To Organisation To Organisation Relationships

*Mr. James Jackson "IN TRUST FOR" Mr. Patrick Jackson "DOING BUSINESS AS" Jackson and Associates Pty. Ltd "TRADING AS" Jacksons International Ltd*

```
<r:Party p:PartyType=”Person”>
 <p:PartyName>
   <n:PersonName>
     <n:NameElement>Mr. James Jackson</n:NameElement>
   </n:PersonName>
 </p:PartyName>
 <!—- Relationship between James Jackson and Patrick Jackson -→
   <r:Relationship r:RelationshipType="IN TRUST FOR"
    r:RelationshipRole="TRUSTEE"
    r:OtherPartyRelationshipRole="BENEFICIARY">
    <r:Party p:PartyType="Person">
     <p:PartyName>
      <n:PersonName>
       <n:NameElement>Mr. Patrick Jackson</n:NameElement>
      </n:PersonName>
     </p:PartyName>
     <!—- Relationship between Patrick Jackson and Jackson and Associates Pty. Ltd -→
     <r:Relationship r:RelationshipType="DOING BUSINESS AS"
      r:RelationshipRole="Director"
      r:OtherPartyRelationshipRole="Company">
      <r:Party p:PartyType="Organisation">
       <p:PartyName>
        <n:OrganisationName>
         <n:NameElement n:ElementType="FullName">Jackson and Associates Pty. Ltd
         </n:NameElement>
        </n:OrganisationName>
       </p:PartyName>
       <!—- Relationship between Jackson and Associates Pty. Ltd and
            Jacksons International Ltd -→
       <r:Relationship r:RelationshipType="TRADING AS"
        r:RelationshipRole="Original Registered Company"
        r:OtherPartyRelationshipRole="Trading Company">
        <r:Party p:PartyType="Organisation">
         <p:PartyName>
          <n:OrganisationName>
           <n:NameElement n:ElementType="FullName">Jacksons International Ltd
           </n:NameElement>
          </n:OrganisationName>
         </p:PartyName>
        </p:Party>     <!—- Jacksons International Ltd -→
      </Relationship>
     </p:Party>       <!—- Jackson and Associates -→
```

```
    </p:Relationship>
   </p:Party>         <!—- Mr. Patrick Jackson -→
  </p:Relationship>
</p:Party>           <!—- Mr. James Jackson -→
```

### 6.11.5 Person To Person To Person Relationships

*Mr. James Jackson is husband of Mrs. Jessie Jackson, and Mrs. Jessie Jackson is the sister of Mr. Craig Smith.*

```
<r:Party>
  <p:PartyName>
    <n:PersonName n:NameKey=”123”>
       <n:NameElement>Mr. James Jackson</n:NameElement>
    </n:PersonName>
  </p:PartyName>
   <r:Relationship r:RelationshipType="Husband-Wife"
    r:RelationshipRole="Husband"
    r:OtherPartyRelationshipRole="Wife">
      <r:Party>
        <p:PartyName>
          <n:PersonName n:NameKey=”456”>
            <n:NameElement>Mrs. Jessie Jackson</n:NameElement>
          </n:PersonName>
        </p:PartyName>
        <r:Relationship r:RelationshipType="Siblings"
         r:RelationshipRole="Sister"
         r:OtherPartyRelationshipRole="Brother">
          <r:Party>
            <p:PartyName>
              <n:PersonName n:NameKey=”789”>
                <n:NameElement>Mr. Craig Smith</n:NameElement>
              </n:PersonName>
            </p:PartyName>
            <r:Relationship r:RelationshipType="Brother-in-Law"
              r:RelationshipRole="Brother-in-Law"
              r:OtherPartyRelationshipRole="Brother-in-Law">
              <p:Party>
                <p:PartyName>
                  <n:PersonName n:NameKeyRef=”123”/>  <!—- Reference Key -→
                </p:PartyName>
              </p:Party>
             </r:Relationship>
          </r:Party>
         </r:Relationship>
       </r:Party>
     </r:Relationship>
 </r:Party>
```

# 7 Differences between two types of Entity Schemas provided by xPRL Specifications

Two types of entity schemas are defined in CIQ V3.0 Specifications and are described in detail in the OASIS CIQ V3.0 Name, Address and Party specification document (see section 3.13). This feature is applicable to this specification also. The two types are:

**Option1 (Default):** All code lists for relationship entity represented using XML schema (in one file) and "included" in the appropriate entity schema (*xPRL-types.xsd*).

**Option 2:** Code Lists represented using Genericode structure of OASIS Codelist TC. Each enumeration list in option 1 is a separate ".gc" file in this option.

## 7.1 Files for Option 1 (The Default)

Following are the additional XML schema files (in addition to the files defined in OASIS CIQ V3.0 Name, Address and Party specification) provided as default in CIQ Specifications package for Option 1:

- *xPRL.xsd*
- *xPRL-types.xsd (**6** Default Code Lists defined for xPRL)*

## 7.2 Files for Option 2

Following is the additional XML schema files (in addition to the files defined in OASIS CIQ V3.0 Name, Address and Party specification) provided as default in CIQ Specifications package for Option 2:

### 7.2.1 XML Schema File

- *xPRL.xsd*

No **-types.xsd* files exist in Option 2 as all the code lists are defined as genericode files.

### 7.2.2 Genericode Based Code List Files

In addition to the files as defined in section 7.2.2 of the OASIS CIQ V3.0 Name, Address and Party specification, following Genericode files are also included.

#### 7.2.2.1 For Party Relationships (xPRL)

6 default genericode based code list files with .gc extension. Each enumeration list in Option 1 is defined as a separate file in Option 2.

## 7.3 Namespace Assignment

Use of namespace for options 1 and 2 is described in the OASIS CIQ V3.0 Name, Address and Party specification document (see section 7.3) are applicable to this specification too.

## 7.4 Differences between CIQ Entity Schemas used in Option 1 and Option 2

Differences between CIQ Entity Schemas used in Option 1 and Option 2 described in the OASIS CIQ V3.0 Name, Address and Party specification document (see section 7.4) are applicable to this specification too.

# 8 Data Exchange and Interoperability

ASIS CIQ TC defines data/information interoperability as follows:

"**Get the *right data* to the *right place* at the *right time* in the *right format* with the *right quality* with the *right security* in the *right context* and with the *right governance* to applications, processes, or users"**

It is the view of the CIQ committee that to enable interoperability of data/information between parties, the best solution is to parse the data elements into its atomic elements thereby preserving the semantics and quality of data. By this way the parties involved in data exchange will be in the best position to understand the semantics and quality of data thereby minimising interoperability issues. How the data will be exchanged between parties, whether in parsed or unparsed structure, must be negotiated between the parties to enable interoperability.

One cannot expect interoperability to occur automatically without some sort of negotiation between parties (e.g. Information Exchange Agreement, whether internal or external to an organisation) involved in data exchange. Once information exchange agreements between parties are in place, then the data/information exchange process can be automated. Moreover, the entire information exchange and interoperability process SHOULD be managed through an effective governance process which SHOULD involve all the parties involved in the information exchange process. This enables effective and efficient management of any change to the information exchange process in the future.

## 8.1 Data Interoperability Success Formula

We at OASIS CIQ TC strongly believe in the following "Data Interoperability Success Formula":

**Data Interoperability = Open Data Architecture + Open Data Integration + Data Quality + Open Data Standards + Data Semantics + Data Security + Data Governance**

All components on the right hand side of the above formula are important for successful data interoperability. The term "Open" used here indicates artifacts that are independent of any proprietary solution (e.g. open industry artifacts or artifacts that are open within an enterprise).

## 8.2 Information Exchange Agreement – Guidelines

To ensure interoperability of CIQ represented data/information between applications/business systems (whether internal to the organisation or external to the organisation) it is strongly advised that an information exchange agreement/specification for CIQ SHOULD is in place. This agreement/specification SHOULD outline in detail the customisation of CIQ specifications.

Following are the features of CIQ specifications that assist in customisation of the specifications to meet specific application or data exchange requirements, and the details of customisation SHOULD be documented and agreed (if involving more than one party in data exchange) at application/system design time to enable automating interoperability of information/data represented using CIQ specifications at application/system run time:

- List of all elements of CIQ XML Schemas that SHOULD be used in the exchange. This includes details of which elements are mandatory and which elements are OPTIONAL
- List of all attributes of CIQ XML Schemas that SHOULD be used in the exchange. This includes details of which attributes are mandatory and which attributes are OPTIONAL
- The approach that will be used for Code Lists (Option 1 or Option 2)
- The code list values that SHOULD be used for each CIQ code lists. This includes updating the default XML Schemas for code lists (Option 1) with the values to be used and updating the default genericode based code lists (Option 2) with the values to be used. These code list files SHOULD then be implemented by all applications/systems involved in data exchange. If genericode based code list

approach (Option 2) is used, then the XSLTs for value validation SHOULD be generated and implemented by all applications/systems involved in data exchange.

- Whether xLink or Key Reference SHOULD be used to reference party, name or address, and the details
- Whether XML schema SHOULD be extended by using new attributes from a non-target namespace and if so, details of the additional attributes
- Whether business rules SHOULD be defined to constrain the CIQ XML schemas and if so, details of the business rules that SHOULD be implemented consistently by all applications/systems involved in data exchange

Once the agreement is implemented, it is vital that the agreement SHOULD be governed through a governance process to manage change effectively and efficiently. All parties involved in the data exchange process SHOULD be key stakeholders of the governance process.

# 9 Conformance

The keywords “MUST”, “MUST NOT”, “SHOULD”, “SHOULD NOT”, “MAY” and “OPTIONAL” interpreted as described in [RFC2119] are used as the conformance clauses throughout this document.

## 9.1 Conformance Clauses

### 9.1.1 Specifications Schema Conformance

Implementation of xPRL Specification MUST conform to the specifications if the implementation conforms to as stated in section 6.9.

### 9.1.2 Specifications Schema Extensibility Conformance

Implementation of xPRL Specification by extending them MUST conform as stated in section 6.6.

### 9.1.3 Specifications Code List Schema Customisation Conformance

Customisation of the Code List XML Schema for xPRL using Option 1 MUST be well formed. Changes to the default values provided as part of the specifications is OPTIONAL and MAY be modified by the user.

### 9.1.4 Interoperability Conformance

Implementation of xPRL Specification between two or more applications/systems or parties helps achieve interoperability if the implementation conforms to using the agreed conformance clauses as defined in sections 9.1.4.1, 9.1.4.2, 9.1.4.3, 9.1.4.4, 9.1.4.5 and 9.1.4.6.

#### 9.1.4.1 Interoperability Conformance – Using Elements and Attributes

Implementation of elements and attributes of xPRL Schema enables interoperability if the following conditions are agreed by two or more parties involved in data exchange and are met:

1. The OPTIONAL elements in the XML Schema that SHOULD be used for implementation and the OPTIONAL elements in the XML Schema that SHOULD be ignored. See section 8.2.
2. The OPTIONAL attributes in the XML Schema that SHOULD be used for implementation and the OPTIONAL attributes in the XML Schema that SHOULD be ignored. See section 8.2 .

#### 9.1.4.2 Interoperability Conformance – Extending the Schema

Implementation of the xPRL schema by extending it SHOULD be agreed and managed between two or more parties involved in the data exchange and MUST be conformed to in order to achieve interoperability as stated in section 6.6.

#### 9.1.4.3 Interoperability Conformance – Using Code Lists

Implementation of a Code List approach SHOULD be agreed and conformance to the selected approach between two or more parties involved in the data exchange MUST be achieved in order to ensure interoperability and this is stated in section 6.2.

#### 9.1.4.4 Interoperability Conformance – Customising the Code Lists

Implementation of the Code List values SHOULD be agreed between two or more parties involved in the data exchange and MUST be conformed to as agreed in order to ensure interoperability as stated in section 6.2.

### 9.1.4.5 Interoperability Conformance – Customising the Schema

Customisation of the schema SHOULD be achieved by the following ways:

1. Using Code List values
2. Defining new business rules to constraint the schema

Implementation of the above approaches SHOULD be agreed between two or more parties involved in the data exchange and MUST be conformed to in order to achieve interoperability as stated in section 6.10.

### 9.1.4.6 Interoperability Conformance – Data/Information Exchange Agreement

Implementation and conformance of the implementation to the agreed Data/Information Exchange Agreement between two or more parties involved in the data exchange MUST be achieved to ensure interoperability as stated in section 8.2.

# A. Acknowledgements

The following individuals have participated in the creation of version 3.0 of xPRL CIQ specifications and are gratefully acknowledged:

**Participants:**

| | | |
|---|---|---|
| Colin Wallis | New Zealand Government | Voting Member, CIQ TC |
| David Webber | Individual | Voting Member, CIQ TC |
| Fulton Wilcox | Colts Neck Solutions LLC | Voting Member, CIQ TC |
| Graham Lobsey | Individual | Voting Member, CIQ TC |
| Joe Lubenow | Individual | Voting Member, CIQ TC |
| John Glaubitz | Vertex, Inc | Voting Member, CIQ TC |
| Michael Roytman | Vertex, Inc | Voting Member, CIQ TC |
| Ram Kumar | Individual | Chair and Voting Member, CIQ TC |

OASIS CIQ Technical Committee (TC) sincerely thanks the public (this includes other standard groups, organisations and end users) for their continuous feedback and support that helps the TC to work toward improving the CIQ specifications.

OASIS CIQ TC also acknowledges the contributions from other former members of the TC since its inception in 2000.

# B. Documentation and Examples

## Documentation

Although, all schema files are fully documented using XML Schema annotations it is not always convenient to browse the schema itself. This specification is accompanied by a set of HTML files auto generated by XML Spy. Note that not all information captured in the schema annotation tags is in the HTML documentation.

## Examples

Several examples of instance XML documents for *xPRL* schema are provided as XML files. The examples are informative and demonstrate the application of this Technical Specification.

The example files and their content are being constantly improved and updated on no particular schedule.

# C. Revision History

| Revision | Date | Editor | Changes Made |
|---|---|---|---|
| V3.0 WD 01 | 02 December 2007 | Ram Kumar | First Version of Committee Working Draft WD 01 |
| V3.0 WD 02 | 15 February 2008 | Ram Kumar | Revised Version of Committee Working Draft WD 01 incorporating TC comments |
| V3.0 WD 03 | 25 February 2008 | Ram Kumar | Version for approval as Committee Draft 01 and includes TC review comments on WD 02 |
| V3.0 CD 01 | 03 March 2008 | Ram Kumar | TC Approved Committee Draft |
| V3.0 PRD 01 | 03 October 2008 | Ram Kumar | Document for 60 days public review |