

# Design of the XBRL specification

**David Vun Kannon**

KPMG Consulting, LLP, New York, USA  
dvunkannon@kpmg.com

**Yufei Wang**

KPMG Consulting, LLP, New York, USA  
yufeiwang1@kpmg.com

***Abstract:***

*Vocabulary design is driven by several factors, including business requirements, available technology and politics. XBRL, as the leading vocabulary for the global accounting industry, has been strongly influenced by each of those factors. The adoption and implementation of XBRL will enable great changes in many financial processes.*

## Introduction

X4GFR (XML for General Financial Reporting) allows software vendors, programmers and end users who adopt it as a specification to enhance the creation, exchange, and comparison of financial reporting information. Financial reporting includes, but is not limited to, financial statements, general ledger information and regulatory filings such as annual and quarterly financial statements. X4GFR is a component of the XBRL (eXtensible Business Reporting Language) framework. This paper discusses the design of the X4GFR vocabulary and accompanying taxonomy schema vocabulary.

The XBRL design was informed from the beginning by the need to satisfy three distinct kinds of requirements – business requirements, technology requirements, and political requirements.

### **Examples of business requirements:**

The XBRL framework is meant to maximize benefits to all stakeholders that will use its specifications.

The X4GFR specification is intended to benefit three categories of users: financial information preparers, intermediaries in the preparation and distribution process, and users of financial information. There is also a fourth category of beneficiary, the vendors who supply software and services to one or more of these three types of user. The overall intention is to balance the needs of these groups creating a product that provides benefits to all groups.

The needs of end users of financial information will generally have precedence over other needs when it is necessary to make specification design decisions that may be perceived as benefiting one community at the possible expense of another.

X4GFR must be extensible, and will support the emerging XBRL framework without any major revisions. A specification for the XBRL framework does not yet exist.

X4GFR is intended to improve the financial statement product.

It should only comply with, not change or set new accounting standards. However, the XBRL framework should facilitate possible changes in financial reporting over the long term.

X4GFR will provide users with a standard format in which to prepare, exchange, and extract financial information.

X4GFR will facilitate the automated comparison of financial information, accounting policies, notes to financial statements between companies, and other items which users may wish to compare that today are performed manually.

The XBRL framework should facilitate "drill down" to detailed information, authoritative literature, audit and accounting working papers.

X4GFR should include specifications for as much information about the reporting entity as may be relevant and useful to the process of financial reporting and the interpretation of the information.

#### **Examples of political requirements:**

The XBRL framework should support international accounting standards and languages other than the American dialect of English.

There should be no default or other preference shown to English as a language, American accounting standards, currency or units of measure.

The Working group and Steering Committee must reach consensus.

#### **Examples of technical requirements:**

XBRL is an XML based design.

X4GFR should be extensible by any adopter to increase its breadth of applicability, and its design should encourage reuse via incremental extensions.

X4GFR should specify the format of information that would be reasonably expected in an electronic format for securities filings by public entities.

X4GFR focuses on the genuine information needs of the user and adheres to the spirit of reporting standards that deprecate the use of bold, italics, and other stylistic techniques that may be used to distract from the true and fair presentation of financial results.

Therefore, there is no functional requirement that X4GFR documents need to support any particular text formatting conventions.

Obviously there is a certain arbitrariness in allocating requirements to any particular category, and it is not our purpose to try to be precise about these things.

#### **Preliminary analysis**

A preliminary document analysis examined financial statements for several large corporations. This analysis revealed:

The documents contained data and the structural relationships of the data items.

The data items were financial measurements, which carried many attributes such as period of relevance, unit, precision, scaling and business entity.

In general these attributes held for most of the data items, but were overridable.

A significant taxonomic hierarchy of financial concepts was reused in each document.

Superficial qualities of the taxonomy, such as concept naming, varied widely from document to document.

The structure of the documents served multiple purposes, including representing taxonomic relationships, arithmetic relationships and conformance to locale specific presentation constraints.

An example is the choice of presenting current or fixed assets first in a balance sheet.

Another input to the design process was a draft DTD that attempted to capture much of the richness of the example documents, including a variety of presentational issues.

The first decision of the working group was to limit the scope of the specification to non-presentational data representation. This decision is consistent with the design principle of separating presentation and content, which is at the heart of XML's own design philosophy.

The second decision was to expand the target representational space from the example documents used in the preliminary analysis to other examples of financial reporting, such as press releases and data interchange by application software. This decision is consistent with the design principle that a generalization of a problem often leads to a simplification in the form of a solution.

An important result of this decision was that the working group took a step back from the domain of financial reporting and looked at the general question of how to represent graphs of related data items in XML syntax. Significant guidance existed already on this topic. Therefore, the design was able to build upon previous work.

We first modeled the example documents as graphs of items and arcs relating them. The modeling was done in UML, using Rational Rose. We felt it was important to get a clear picture of the domain issues, without the preconceptions or biases of a particular technology target, such as XML. Even though we knew XML was a technical requirement, we realised that modeling the domain using XML artifacts such as DTDs or X Schemas (then very preliminary) would skew the domain model because of gaps in the expressive power of these languages. This is as clear an example as could be asked for of the Whorfian Hypothesis at work .

The working group realised that, while the items represented instance data that was irreducibly necessary, the arcs represented structural information that was reused from document to document. Further, business users required the use of consistent structure across documents, and the work of other business users would be advanced if structure could be discussed separately from instances of the structure.

Therefore, a very significant decision was taken to separate the data items into instance documents and the structural information into a separate document, which would be referred to by the instance documents. These specifications of structural information have come to be called taxonomies within the working group. This decision allowed the working group to see the specification of each component as separate work items that could be pursued in parallel.

## Syntax of instance documents

The syntax chosen for X4GFR instance documents corresponds to a recommended syntax for serializing graphs of data in XML . X4GFR uses this canonical syntax to exploit the features of XML attributes, specifically their order independence, irredundancy, ability to accommodate enumerated types, and the ability to have default (#IMPLIED) values. In X4GFR there are relatively few XML elements, but there is a rich set of attributes that are applicable to most elements, and furthermore, the allowed values for those attributes are elements within one or more taxonomies.

### Common attributes

The core syntax for statements is defined using an XML DTD. The elements defined there are the item, label, and group. The item and group elements have the same set of attributes, which are in some sense the more important part of the X4GFR representation. The set of attributes is defined as follows.

```
<!ENTITY % att_AttributeHolder "  
    id                CDATA          #IMPLIED  
    period            CDATA          #IMPLIED  
    schemaLocation    CDATA          #IMPLIED  
    scaleFactor       CDATA          #IMPLIED  
    precision         CDATA          #IMPLIED  
    type              CDATA          #IMPLIED  
    unit              CDATA          #IMPLIED  
    entity            CDATA          #IMPLIED  
    decimalPattern    CDATA          #IMPLIED  
    formatName        CDATA          #IMPLIED  
">
```

The set of attributes given above is meant to make explicit the answers to such questions as "When?, How Many?, For Whom?, According to Whom?" etc. Without reliable answers to these questions, a financial measurement has no force.

Each attribute is described and discussed separately below.

### #IMPLIED resolution

If an attribute that is specified as #IMPLIED is not present in an instance of an item, it must be available attached to a container element that is an ancestor (in the XPath sense) of the item element. The id attribute is the obvious exception; an item without an id does not inherit one from its containing parent.

The XPath expression "ancestor-or-self::\*[@implied-attribute][1]/@implied-attribute" finds the nearest value of an attribute, which may appear either attached to an element of higher up in the document tree.

Note:

The working group looks forward to redefining the instance syntax using XSchema. One key benefit of this will be the ability to document our #IMPLIED resolution strategy precisely in the machine-readable version of the specification.

The implication of this is that an X4GFR item element is always fully specified in terms of all of its attributes, even if some of those attributes are not directly attached to the item element itself. There are no default values for any of the attributes that can appear on an item or group, yet the specification mandates that a value be available for these attributes at each node in the instance document.

This is a key requirement for full internationalization. Every X4GFR instance document must specify, for all items, all relevant attributes. In particular, item types do not default to US dollars, US measurements, US number formatting conventions, US accounting principles, or US English for anything other than the names of elements and attributes.

### id

This attribute is not required, since the use of XPointer is encouraged for referring to specific elements in an X4GFR instance document. The content must start with an alpha character.

<b>Example</b>	<b>Meaning</b>
C2424	A unique identifier

**Table 1**

An early decision was to anticipate the success of the whole constellation of XML related standards, rather than just the XML 1.0 Recommendation itself. Therefore, the use of namespaces, XSL, XPointer, and XSchema was encouraged, since these Recommendations bring significant expressive power to bear in a standard way.

### **period**

Every item applies to a particular instant or duration. This attribute uses the ISO 8601 date representation. A duration is a pair of dates separated by a slash.

<b>Example</b>	<b>Meaning</b>
1999	The full year 1999.
2000-04	The month of April 2000.
1999-12-31	The last day of 1999.
1999-12-25/2000-03-25	The three months ended 25 March 2000.
2000-01-01/2000-04-01	The thirteen weeks ended 1 April 2000.
1999-12-29/2000-03-27	Period from December 29, 1999 through March 27, 2000.
2000-01/2000-03	The first three months (first quarter) of 2000.

**Table 2**

See <http://www.w3.org/TR/NOTE-datetime> for more information.

The period attribute is different from several other attributes in that it mandates a specific date format, instead of providing a framework for representation choice. While the working group recognised that there are many calendars in use throughout the world, and many date representation systems, it was considered more important to follow the lead of the broader Internet community as represented by the above referenced Note.

### **entity**

An entity specifies a system for identifying business entities and a particular identifier within that system. A business entity does not have to be a full corporate entity; it could be a subsidiary, a division, even an individual; any organization for which there is a financial statement.

<b>Example</b>	<b>Meaning</b>
NASDAQ:SAMP	The company with NASDAQ ticker symbol SAMP.
CUSIP:41009876AB	The entity CUSIP number 41009876AB (e.g. a mutual fund).
URI: www.w3c.org	The non-profit organization owning the URI <a href="http://www.w3c.org">http://www.w3c.org</a> .

**Table 3**

The entity is a QName so as to provide a framework for referencing naming authorities. It does not imply that the AICPA is a naming authority for business entities.

Note:

The attributes that specify Qnames as content follow the lead of the XSchema specification in this regard. When the instance syntax is further specified by an XSchema document, the datatype of these attributes will be refined to QName from the current CDATA in the DTD.

## **type**

The type attribute provides the name of an element within a taxonomy. Its purpose is to define the financial concept, which is being measured. The use of standard concepts will facilitate comparison of information contained in different instance documents. A convention followed in the AICPA USGAAP C&I taxonomy is that the name of a type is a dot-separated pair of camel-case identifiers representing a human readable name for the concept and its parent.

The reason for the "parent.child" naming convention is that within a taxonomy, it is important for an element name to be unique. A single name such as "NetIncome" is inadequate because it could appear at multiple points in a taxonomy. Adopting the (parent.child) naming convention helps, but still turns out to be no guarantee. An extreme solution, which was discarded early in the design, would be to use a number. It was felt that there was nothing to be gained in a numeric identifier.

<b>Example</b>	<b>Meaning</b>
aicpa:CurrentAssets.Receivables	Receivables
aicpa:AccountantsReport.ReportingMethod	Reporting Method
aicpa:NotesToFinancialStatements.GoingConcern	A "Going Concern" note

**Table 4**

Note that the type is a QName. The type attribute content must contain the correct namespace prefix, and all namespaces used in attributes must be declared in the document instance. However, it is still necessary to have a schemaLocation attribute to attach the namespaces to the right files.

## schemaLocation

The schemaLocation attribute is used to hold the base URI of the taxonomy that includes the concept of which the item is an instance.

Note that the way that namespaces are defined in XML, there is no guarantee that a URI with which a namespace is associated can be dereferenced to something useful. For example, the attribute `xmlns:NASDAQ="http://www.nasdaq.com/XBRL/ticker"` does not imply that any such URI actually points to any service having to do with ticker symbol lookup. XML Schema defines the schemaLocation attribute (<http://www.w3.org/TR/xmlschema-1/#xsi:schemaLocation>) that can be used in a document to provide hints as to the physical location of schema documents to be used for validation. There is further discussion of this in the XML Schema Primer . Because this is exactly the purpose intended for this attribute in XBRL, the same attribute name has been used.

Example	Meaning
<a href="http://www.iasc.org/x4gfr/airline/00-07-07">http://www.iasc.org/x4gfr/airline/00-07-07</a>	A taxonomy for the airline industry conforming to IASC guidelines.
<a href="http://www.x4gfr.org/aicpa/us/gaap_ci/00-03-3">http://www.x4gfr.org/aicpa/us/gaap_ci/00-03-3</a>	The AICPA's US GAAP taxonomy for commercial and industrial companies.
<a href="http://www.x4gfr.org/cica/canada/media/00-06-02">http://www.x4gfr.org/cica/canada/media/00-06-02</a>	A Canadian Institute of Chartered Accountants' taxonomy for media companies.

Table 5

By publishing a taxonomy structure for US GAAP, the AICPA hopes to facilitate the comparability of data from many sources. However, creators of X4GFR data may refer to specific authoritative sources via the schemaLocation attribute, rather than defining the AICPA as the only source for taxonomy. Business entities, governments, software vendors, standards bodies and auditors can all create taxonomic resources that are publicly referencable. The voluntary extension and refinement of published taxonomies will allow for the flexibility in reporting concepts that most users of X4GFR require, especially in the international arena.

While not strictly necessary, the existence of this attribute separate from the type attribute leads to considerably cleaner (shorter and more readable) files. Ideally, the namespace declarations, which are referenced by the prefix parts of the Qualified Names of the type attribute content, would be sufficient to connect that content to the schema document. Unfortunately, that does not seem possible to enforce.

## unit

Unit specifies the standard that is relevant to the measurement. It is expected that most measurements will be monetary measurements. ISO 4217 standard currency designation is required for the units attribute in such a case. (<http://www.iso.ch/cate/d23132.html>) Dimensional measurements should be given in the SI system. Pure numbers and counts of people, shares and the like can be specified as quantities. Enumerations depend on the taxonomy in force for the item's concept to specify the datatype of the element as an enumerated datatype, and to provide the allowable values.

<b>Example</b>	<b>Meaning</b>
ISO4217:GBP	Currency, UK Pounds.
SI:m2	Square meters
ISO4217:USD	Currency, US Dollars
US:ft2	Square feet
Moodys:rating	Credit rating
AICPA:pure	A pure number

**Table 6**

Since unit is defined as a Qualified Name, the prefixes in the above examples must have been previously defined in namespace declarations.

### **scaleFactor**

An integer power of ten. If a scale value is not 0 (the default), the numeric value of the item element must have the proper multiplier applied to arrive at the actual value.

<b>Example</b>	<b>Meaning</b>
3	Thousands
-6	Millionths

**Table 7**

### **precision**

Precision is an integer intended to convey the arithmetic precision of a measurement, and therefore, the utility of that measurement to further calculations.

<b>Example</b>	<b>Meaning</b>
9	Precision of nine decimal digits.

**Table 8**

The previous two attributes, and the next two as well, apply only to content with monetary or similarly numeric datatype. This is the vast majority of content for XBRL documents.

### **decimalPattern**

decimalPattern is used to hold locale specific formatting for the value, precision, and scale attributes. It follows the usage of the XSLT Recommendation, Section 12.3 - Number Formatting. It corresponds to the second argument of the number-format function. For more information see the source documents:

<http://java.sun.com/products/jdk/1.1/docs/api/java.text.DecimalFormat.html>

<http://java.sun.com/docs/books/tutorial/i18n/format/decimalFormat.html>

Example	Meaning
#,###.##	Typical US numbers, default treatment of negative with a leading minus sign.
#,###.##;(#,###.##)	Comma separator every three digits, negative numbers in parentheses and no minus sign.
#,###,##	Comma used as decimal separator. (Needs a formatName attribute for correct interpretation.)

Table 9

Inclusion of this and the following attribute in the specification is intended to allow for the use of X4GFR in international settings. The referenced XSLT Recommendation of the W3C itself refers to the JDK 1.1 specification for the details of constructing number formats. This reference to the JDK is not meant as requirement to use the JDK or Java in the implementation of applications that will use X4GFR; rather, it merely references a widely available source of information.

### formatName

formatName refers to an element from an XSLT namespace that is used to define a decimal format. It follows the usage of the XSLT Recommendation, Section 12.3 - Number Formatting. It corresponds to the third argument of the number-format function. If present, the document containing the item should also contain a decimal-format element from the XSLT namespace whose name matches the content of this attribute.

```
<xsl:decimal-format
  name = qname
  decimal-separator = char
  grouping-separator = char
  infinity = string
  minus-sign = char
  NaN = string
  percent = char
  per-mille = char
  zero-digit = char
  digit = char
  pattern-separator = char
/>
```

See <http://www.w3.org/TR/xslt#format-number> for more information.

decimalPattern and formatName are two attributes that the working group believed necessary to support internationalization. Rather than duplicate work done in the JDK and XSL, the specification leverages that work directly.

### Elements

#### The item element

As discussed above, an item represents a single fact or statement within a report. Although the content model of item allows parsed character data, the value is actually further restricted by the datatype given to the item type in the taxonomy.

```
<!ELEMENT item (#PCDATA )>
<!ATTLIST item %att_AttributeHolder; >
```

Example	Meaning
<pre> &lt;item entity="SAMP" period="19981231" xmlns="http://www.x4gfr.org/core/00-03-31" xmlns:aicpa="http://www.x4gfr.org/aicpa/us/gaap/ci/03-03-31" schemaLocation="http://www.x4gfr.org/aicpa/us/gaap/ci/03-03-31.xsd" type="aicpa:LongTermAssets.NetCapitalLeases" unit="ISO4217:USD" scaleFactor=3 precision=3 decimalPattern="#.###" formatName=""/&gt;727 &lt;/item&gt; &lt;item entity="SAMP" period="19981231" xmlns="http://www.x4gfr.org/core/00-03-31" xmlns:aicpa="http://www.x4gfr.org/aicpa/us/gaap/ci/03-03-31" schemaLocation="http://www.x4gfr.org/aicpa/us/gaap/ci/03-03-31.xsd" type="aicpa:LongTermAssets.NetCapitalLeases" unit="ISO4217:USD" scaleFactor="3" precision="3" decimalPattern="#.###" formatName="" xmlns:ati="http://www.x4gfr.org/core/00-03-31"/&gt; </pre>	<p>Net Capital Leases at calendar year end 1998 for SAMP of \$727,000.</p>
<pre> &lt;item entity="SAMP" period="19981231" xmlns="http://www.x4gfr.org/core/00-03-31" xmlns:aicpa="http://www.x4gfr.org/aicpa/us/gaap/ci/03-03-31" schemaLocation="http://www.x4gfr.org/aicpa/us/gaap/ci/03-03-31.xsd" type="aicpa:NotesToFinancialStatements.Concentrations" /&gt; Concentration of credit risk with regard to short term investments is not considered to be significant due to the Company's cash management policies. These policies restrict investments to low risk, highly liquid securities (that is, commercial paper, money market instruments, etc.), outline issuer credit requirements, and limit the amount that may be invested in any one issuer. &lt;/item&gt; </pre>	<p>The text of the Concentrations note at calendar year end 1998 for SAMP.</p>

**Table 10**

The item element should be regarded as a leaf in the tree of X4GFR elements within a given instance document. In particular, the content of an item should not contain other items. The design considered at one time allowing items to contain items, which would obviate the need for a group element. After discussion, this was felt to be an overloading of responsibilities on the element.

Items are the factual residue, the irreducible content of an instance.

### **The group element**

The group element aggregates attributes for items, so that attributes do not have to be given in full on each item. The group element also forms a simple container element for multiple items in a document

instance. Items inherit the value of attributes from the closest parent with an explicit reference to the attribute value. The group element provides a convenient way to group similar items together, without forcing a particular hierarchy. Entity, period and type are all useful grouping attributes, and the specification allows each document to use them in whatever order is desired.

```
<!ELEMENT group (item | group | label)*>
<!ATTLIST group %att_AttributeHolder; >
```

Example	Meaning
<pre>&lt;group entity="SAMP" type="LongTermAssets.NetCapitalLeases" xmlns="http://www.x4gfr.org/core/00-03-31" xmlns:aicpa="http://www.x4gfr.org/aicpa/us/gaap/ci/03-03-31" schemaLocation="http://www.x4gfr.org/aicpa/us/gaap/ci/03-03-31.xsd" unit="ISO4217:USD" scaleFactor=3 precision=3 decimalPattern="#,###" formatName=""&gt; &lt;item period="19981231"&gt;635&lt;/item&gt; &lt;item type="NotesToFinancialStatements.Concentrations" period="19981231"&gt; Concentration of credit risk with regard to short term investments is not considered to be significant due to the Company's cash management policies. &lt;/item&gt; &lt;/group&gt;</pre>	<p>Use of group with items varying by period and type:</p>
<pre>&lt;group entity="SAMP" xmlns="http://www.x4gfr.org/core/00-03-31" xmlns:aicpa="http://www.x4gfr.org/aicpa/us/gaap/ci/03-03-31" schemaLocation="http://www.x4gfr.org/aicpa/us/gaap/ci/03-03-31.xsd" unit="ISO4217:USD" scaleFactor=3 precision=3 decimalPattern="#,###" formatName=""&gt; &lt;group period="19981231"&gt; &lt;item type="LongTermAssets.NetCapitalLeases"&gt;727 &lt;/item&gt; &lt;item type="aicpa:NotesToFinancialStatements.Concentrations"&gt; Concentration of credit risk with regard to short term investments is not considered to be significant due to the Company's cash management policies. &lt;/item&gt; &lt;/group&gt; &lt;item type="LongTermAssets.NetCapitalLeases"&gt;635 &lt;/item&gt; &lt;/group&gt; &lt;/group&gt;</pre>	<p>Use of group with nesting by period, and items within the groups varying by type.</p>

Table 11

The significance of these two examples is that the "group" container element is designed to allow for more compact instance documents. It is not intended to be a general structuring mechanism intended to convey presentation related information. Applications should not produce instance documents with group elements expecting all consuming applications to respect those groupings, their type, label or other attributes for any other usage than to assign attribute values to all their contained elements.

## The label element

All elements within a given taxonomy have a label assigned to them in one or more languages. The label element allows applications to override that label.

```
<!ELEMENT label(#PCDATA)>
<!ATTLIST label href CDATA #IMPLIED >
```

Note that the href attribute is CDATA, not IDREF. Producing applications should create documents that use XPointer instead of IDREF. If the href content is not href="xpointer(...)" then a consuming application can try to interpret it as an IDREF to an item with an id attribute, but the DTD/XSL/DOM hooks will not work correctly, e.g., it will not be possible to use the XSL/XPath id() function.

Note also that although <label> is legal in instance documents, it is really intended for use in taxonomy documents. Occurrence of label elements in the instance document is a last resort. If a company has a particular style of rendering a common accounting concept, that should be held in an extension taxonomy for that company. Labels in instance documents apply to that document only, which implies a very temporary usage.

Example	Meaning
<pre>&lt;group&gt; &lt;item type="aicpa:Assets.CurrentAssets"&gt;657&lt;/item&gt; &lt;label href="xpointer(//item[@type="aicpa:Assets.CurrentAssets"&gt;657])" /&gt; Capital Assets&lt;/label&gt; &lt;/group&gt;</pre>	<p>Every instance of an item with type="aicpa:Assets.CurrentAssets" gets the new label instead of the one in the taxonomy that was in the Working Draft providing it.</p>

Table 12

## Overall instance document design rationale

Some of the features used in X4GFR instance documents appear to be at odds with conventional definitions of XML document types. Order independence and the heavy use of attributes are relatively novel but offer crucial advantages to meet both defined current requirements and known future requirements.

### Order independence

Although the ordering of financial information in presentation to a human is important, ordering is irrelevant insofar as the exchange of data between software applications is concerned. Therefore, in XBRL the ordering of item elements is unimportant and there is no document structure defined within the core specification. The main reason for this is that it greatly increases modularity. For one thing, it allows any XML document in the world that happens to describe financial information to include an X4GFR item to describe it. Consider an HTML press release with an embedded X4GFR item:

Example	Meaning
---------	---------

<P>San Sushi, CA, 7 December 2002: Sample X4GFR embedded in HTML.  
 Data Inc. today announced net revenues of  
 \$<item  
 xmlns="http://www.x4gfr.org/core/00-03-31"  
 xmlns:aicpa="http://www.x4gfr.org/aicpa/us/gaap/ci/03-  
 03-31"  
 schemaLocation="http://www.x4gfr.org/aicpa/us/gaap/ci/03-  
 03-31.xsd" entity="SAMP"  
 type="aicpa:IncomeStatement.Revenues"  
 precision="2" scaleFactor="6"  
 decimalPattern="#.##"  
 period="20020701/20021001"  
 units="USD">7.2</item>m for the third quarter  
 of 2002.</P>

**Table 13**

There are many reasons for wanting this embeddability feature; support for XML-aware search engines, and embedding X4GFR items within the documents of other electronic commerce protocols are only two. As a historical note, this was one of the earliest decisions of the XBRL specification working group (Chicago, October 1999), to factor out the structure into a separate document, instead of burdening every instance document with its repetition.

Order independence also simplifies the combination of financial information from different periods or entities, or even for the same entity under different reporting regimes, since in most cases an X4GFR instance document can be created by concatenating other X4GFR instance documents.

Finally, order independence makes it easier for individual reporting entities to define incremental new types, labels in different languages, etc. This level of extensibility is known to be a key requirement for meeting the reporting needs of a substantial number of business entities and has been an overriding consideration in the design of the language.

### **Use of attributes**

The interpretation of most items is embedded within attributes. Applications that process the information in an X4GFR document, such as (say) rendering as HTML will process each item and perform a "lookup" into the appropriate taxonomy in order to extract properties such as its appropriate text label in a given human language, to determine the order in which it should be presented in a table relative to other items, etc. In principle, all of the attributes of items (except id) could have been done as an optional sub element. However, this would have sacrificed the ability to rely on the semantics of #IMPLIED attributes, as well as making the document instance unnecessarily verbose.

The use of a general group construct to aggregate content for subordinate attributes, means that pure X4GFR documents resemble a database more than they resemble a presentation-oriented document. This is intentional, since future extensions of X4GFR into the arena of internal reporting will in fact require X4GFR to serve, in effect, as a neutral format for passing multidimensional data from one application to another. Again, this was one of the earliest decisions of the XBRL specification working group (Chicago, October 1999), to use a minimal set of elements and to exploit attributes as the preferred way of establishing a rich semantic context for any given financial number.

That value should be carried as element content rather than as an attribute was a fairly late decision, reflecting the preference of implementors to avoid carrying potentially long strings of text as attribute content, and to avoid creating to essentially identical elements, one with a value attribute for numeric content, the other with element content for strings.

## **Syntax of taxonomy documents**

### **Taxonomy design rationale**

There are several strengths displayed by the current design of X4GFR taxonomies.

First, it leverages existing standards, specifically, XML and those portions of XML Schema that are very unlikely to change.

Second, it supplies a commonly used set of elements (in this case, US GAAP for Commercial and Industrial companies) and therefore does not require every document instance to be designed "from scratch".

Third, it is fully extensible.

Alternative approaches to extensibility were considered and failed to meet the requirements for complete independence from language, set of accounting principles, and document types. This required a novel use of XML Schema not as a replacement for DTDs, but as a kind of concept definition language.

Before selecting XSchema as the foundation for taxonomy documents, several other ontology markup languages were considered. Several factors contributed to the selection of XSchema

XSchema was relatively simple to understand.

XSchema was the product of a large effort within the Internet community.

Other markup languages were supported by single academic researchers.

XSchema was more likely to be supported by third party software.

### **Syntax of taxonomies**

Although only one X4GFR taxonomy exists as of the release of this specification, there will be many. Each taxonomy consists of a list of new element definitions along with relations between these elements. The definition of a taxonomy is done by extending the XML schema using several linking structures that will be described here. The syntax of a taxonomy is defined in a metamodel that contains definitions for certain simple and complex data types, each of which will be described here.

#### **The monetary and pure datatypes**

The X4GFR metamodel defines a datatype "monetary" that specializes the "decimal" type. Monetary strings are interpreted with respect the enclosing decimalPattern for any item where they appear. A taxonomy, which includes numeric elements that are meant to be interpreted as monetary values, should use this datatype rather than "string", which is the default.

The empty string "" is not a valid instance of the monetary datatype.

A negative number is a valid instance of the monetary datatype. Any item with datatype "monetary" can have a negative number as its value. The presentation of negative numbers (often in parentheses) is relatively rare in financial reporting; it is the responsibility of the producing application to ensure that the sign of the number indeed indicates a negative balance, i.e., negative with respect to the normal balance for a given type.

The datatype "pure" also specializes the "decimal" type. While items with "monetary" datatype should have a measure from the ISO 4217 namespace of currencies, items with "pure" datatype are to be used for pure numbers such as ratios, percentages, and number of shares, that is, financial numbers other than amounts of money.

## element

An element has a name and data type. Because the US GAAP C&I Taxonomy contains several hundred items, a method was needed to prevent name clashes and this led to the convention of using names such as "MarketableSecurities.AvailableForSale" containing both the colloquial name of the item as well as its immediate parent in the taxonomy. This convention is not a requirement of any taxonomy, although it is the case that all element names must be unique within a given taxonomy.

Example	Meaning
<pre>&lt;element name="Dividends.Preferred" type="monetary"/&gt; &lt;element name="NotestoFinancialStatements.StockBasedCompensation" type="string"/&gt;</pre>	From US GAAP C&I Taxonomy

Table 14

The usage of the "element" element in an X4GFR taxonomy is syntactically no different that its usage in XML Schema. For those familiar only with DTDs, these definitions are equivalent to use of a definition such as `<!ELEMENT Dividends.Preferred>`, with the additional power of Schema Constraints, in this case, data type constraints. This does not mean, however, that X4GFR instance documents should be construed as containing forms such as `<Dividends.Preferred>` as structural elements; they do not.

The US GAAP C&I Taxonomy includes several hundred element definitions.

## links

The links element is a container for rollup and label elements in a taxonomy. Elements do not need to be within the scope of the links element.

```
<complexType name="LinksType">
  <element name="rollup" type="RollupType"/>
  <element name="label" type="LabelType"/>
</complexType>
<element name="links" type="LinksType"/>
```

Example (this is the structure of the US GAAP C&I Taxonomy schema file):

**Example**

```

<schema targetNames-
pace="http://www.x4gfr.org/aicpa/us/gaap/ci"
xmlns="http://www.x4gfr.org/aicpa/us/gaap/ci">
<element
name="Statements.ManagementDiscussionandAnalysis"
type="string"/> <element
name="StatementInformation.Reportingentity"
type="string"/> <links> <rollup
from="ManagementDiscussionandAnalysis.ForFullFiscalYear"
to="Statements.ManagementDiscussionandAnalysis"
sense="add" order="1"/> <label
href="StatementInformation.Reportingentity"
xml:lang="en">Reporting entity</label>
</links> </schema>

```

**Meaning**

From US GAAP C&I Taxonomy

**Table 15**

**Note:**

We anticipate that in the future, the elements contained in the links section of a taxonomy will be refactored, so that the individual rollup and label elements will be contained within annotation and appinfo elements that are children of the element elements. We await better documentation of these features of the XSchema specification before putting them to use. When this refactoring is accomplished, it will be easier to relate elements, rollups and labels.

**rollup**

The rollup element defines how elements are related to one another in a parent-child relationship. The actual declaration within the X4GFR metamodel defines a RollupType, with the rollup element being an element of that type.

```

<complexType name="RollupType">
  <attribute name="from" type="QName"/>
  <attribute name="to" type="QName"/>
  <attribute name="sense" type="string">
    <enumeration value="add"/>
    <enumeration value="subtract"/>
    <enumeration value="none"/>
  </attribute>
  <attribute name="order" type="decimal"/>
  <attribute name="reference" type="string"/>
</complexType>

```

There are four required attributes, from, to, sense, and order, and one optional attribute, reference.

**from**

A qualified name that indicates the child element in the relation.

**to**

A qualified name that indicates the parent element in the relation.

**sense**

Indicates whether the relationship is simply one of inclusion (e.g., discussion of interim periods is part of the management discussion and analysis, and both are text), or is an arithmetic relationship and whether its mathematical sense is additive or subtractive. Enumerated values are none, add, and subtract.

**reference**

Reference is a text string that provides a human readable reference to supporting accounting or other literature. It is a business requirement and goal of the XBRL framework to provide drill-down abilities to enable reference back from instance documents to the accounting literature. The reference attribute is only a start in this direction. Much of the authoritative accounting literature, in the United States and other countries, is not available on-line from the issuing authority. Therefore, it is not possible to point directly into the relevant documents using Xlink and Xpointer technology.

**order**

Order is a nonnegative decimal number that indicates how sibling elements are normally ordered for presentation within their parent element. A consuming application is in principle free to ignore this order parameter. Presentation order was one of the commonalities that the specification working group realised could be factored out of instance documents into the taxonomy specifications. Ordering is often tied to a particular taxonomy, such as the ordering differences in the balance sheet between the US and the UK.

Note that order is a decimal, not an integer. It is possible for a taxonomy to extend another, inserting a new element in between existing elements, using a decimal number of the necessary precision.

Example	Meaning
<pre>&lt;rollup from="ManagementDiscussionandAnalysis.InterimPeriods" to="Statements.ManagementDiscussionandAnalysis" sense="none" order="2"/&gt; &lt;rollup from="BalanceSheet.Networkingcapital" to="FinancialHighlights.BalanceSheet" sense="add" order="3"/&gt; &lt;rollup from="AccountsReceivable- Tradenet.AllowanceforDoubtfulAccounts" to="Receivables.AccountsReceivable-Tradenet" sense="subtract" order="2"/&gt;</pre>	From US GAAP C&I Taxonomy

Table 16

The US GAAP C&I Taxonomy includes several hundred rollup elements.

The syntax of the "from" and "to" elements is essentially shorthand for XPointer syntax, for example:

Example	Meaning
<pre>&lt;rollup from=xpointer(element[@name="BalanceSheet.Networkingcapital"]) to=xpointer(element[@name="FinancialHighlights.BalanceSheet"]) sense="add" order="3"/&gt;</pre>	Full XPointer syntax

Table 17

## label

One of the key internationalization features of X4GFR is that although each taxonomy defines a single set of elements representing a coherent set of accounting concepts, the label<sup>34a</sup> string used to present the name of that concept<sup>34</sup>is declared separately with an indication of the language using the XML standard lang attribute. Thus, a given set of financials could be presented by a single application in a language selected by the user (although recasting the underlying financials under a different set of national accounting principles is a far more complex matter).

```
<complexType name="LabelType" content="textOnly">
  <attribute name="href" type="QName"/>
  <attribute name="xml:lang" type="language"/> </complexType>
```

Example	Meaning
<pre>&lt;label href="BalanceSheet.Cashandcashequivalents" xml:lang="fr"&gt;Argent Comptant&lt;/label&gt; &lt;label href="BalanceSheet.Cashandcashequivalents" xml:lang="en"&gt;Cash and cash equivalents&lt;/label&gt;</pre>	From US GAAP C&I Taxonomy

Table 18

These definitions can be in separate files; ordering is unimportant. Labels can be overridden in an instance document. The latter feature is important because individual companies routinely adjust the wording of an otherwise standard category, to reflect their particular circumstances.

## Acknowledgments

Thank you to my Co-Chair of the Specification Working Group, Walter Hamscher of PricewaterhouseCoopers, for many insightful discussions, and for his editing of the specification. Thank you to all members of the Specification Working Group for their questions, contributions and criticisms, which led to a better final spec.

## Bibliography

- [CANONICAL] Bosworth, A., A. Layman and M. Rys. Serializing Graphs of Data in XML. BizTalk.org Library, Microsoft Corporation, 1999.
- [WHORF] Whorf, B. L. (1940): 'Science and Linguistics', *Technology Review* **42**(6): 229-31, 247-8. Also in B. L. Whorf (1956): *Language, Thought and Reality* (ed. J. B. Carroll). Cambridge, MA: MIT Press
- [SCHEMA-0] World Wide Web Consortium. XML Schema Part 0: Primer.

## Authors

### David Vun Kannon

Manager  
KPMG Consulting, LLP  
Postal Address:  
757 3rd Ave, Suite 1401  
10017 New York  
NY  
USA  
Telephone: +1 212 872 7713  
Fax: +1 212 954 2744  
E-mail: dvunkannon@kpmg.com

**David vun Kannon** - David vun Kannon is a manager in the Financial Services practice of KPMG Consulting. He is responsible for the information architecture for projects developed within Financial Services Consulting that highlight the technology expertise of KPMG Consulting and the close relationship the firm has with strategic partners in Electronic Commerce applications for the Financial Services line of business. Recent accomplishments include leading the technical development of the XBRL specification and advisement to the Global Straight Through Processing Association (GSTPA) on the creation of an object model and resulting XML vocabulary to support the messaging requirements of that organization. The GSTPA is an organization of the most technologically advanced securities firms, and includes representatives of global standards-making bodies. This XML vocabulary will define the structure and content of all messages involving the settlement of cross-border securities transactions handled by the GSTPA.

### Yufei Wang

Consultant  
KPMG Consulting, LLP  
Postal Address:  
757 3rd Ave, Suite 1401  
10017 New York  
NY  
USA  
Telephone: 212 954 6358  
Fax: +1 212 954 2744  
E-mail: yufeiwang1@kpmg.com

**Yufei Wang** – Yufei Wang is a consultant in the KPMG Consulting Electronic Commerce practice. She is a member of the Solution Integration team in the Financial Services Consulting area. As a member of the SI team, Yufei is responsible for the design and development of applications and architecture for Financial Services clients as well as internal thought leadership initiatives. Yufei is an expert in XML, XSL technologies, and the application of UML-based modeling techniques to enterprise architecture, messaging, and DTD design.