

O-030

## RELAX NG による UBL スキーマライブラリの開発 Development of UBL Schema Library in RELAX NG

衣川 勇樹 †  
Yuuki Kinugawa

内藤 広志 ‡  
Hiroshi Naito

### 1 はじめに

B2B において業界を問わず利用可能なビジネス文書のライブラリである Universal Business Language (UBL) が開発されている[1]。UBL は XML のビジネス文書を作成する際に使用する共通の部品を W3C XML Schema (XSD) で定義している。今回 XSD で記述された UBL スキーマを、シンプルでツール開発が容易なスキーマ言語である RELAX NG (RNG) [2] で記述するため、UBL の XSD スキーマを RNG スキーマに変換するツールを開発した。

本発表は [3] の研究を元に、最新版の UBL へ対応させ、さらに XSD の派生型の変換とスキーマ情報の追加を行った結果を述べたものである。

### 2 UBL の構成

UBL は以下の 3 つを提供している。

- (1) スキーマライブラリ：ビジネス文書の部品（コアコンポーネント）を定義するための XSD で記述されたスキーマ群
- (2) ビジネス文書：(1) のスキーマを組み合わせで定義された受発注用の XML 文書
- (3) 拡張用のガイドライン：(1) のスキーマをカスタマイズし、業界特有の文書を作成するための規定

(1) は次の 8 種類のスキーマから構成されており、図 1 のような参照関係にある。このスキーマには複数個のスキーマが存在する。

CoreComponentParameters：コンポーネントの注釈を記述  
CoreComponentTypes：コアコンポーネントの型を定義  
UnspecializedDatatypes：を派生させ、ビジネス文書で使用するデータ型を定義  
SpecializedDatatypes：ビジネス文書で使用する付加的なデータ型を定義  
CommonBasicComponents：ビジネス文書に使用する基礎的な共通コンポーネントを定義  
CommonAggregateComponents：を集約してビジネス文書に使用する共通コンポーネントを定義  
codelist：通貨コードなど、様々なコードを選択するリストを定義（13 個）  
maindoc：各コンポーネントを組み合わせ、標準的なビジネス文書を定義（8 個）

図 1 のとおり、下位のスキーマはより上位のスキーマによって参照され、最上位の maindoc スキーマは(2)の XML 文書を定義する。なお、このスキーマは全てのスキーマが

参照しているため図 1 では省略した。

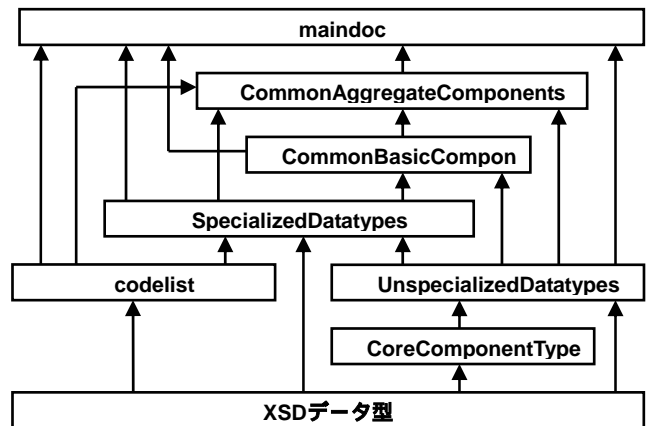


図 1 スキーマの参照関係

### 3 RNG スキーマライブラリの開発

#### 3.1 開発の目的

現在 UBL が提供しているスキーマライブラリは XSD で記述されている。XSD は W3C が勧告したスキーマ言語で、多くの要素や属性を持ち、高い表現力を持っている。しかし、それゆえ構造が複雑で、理解しにくいという欠点もある。そこでスキーマライブラリを RNG に変換するツールを開発した。RNG は OASIS[3] によって標準化されたスキーマ言語で、シンプルで柔軟な構造を持つ。RNG でスキーマライブラリを記述することで、RNG を使用したツールの利用を可能にし、UBL の開発の幅を広げることができる。

#### 3.2 開発の方法

図 2 に示すとおり、RNG スキーマライブラリは XSD スキーマライブラリをツールで変換したものである。ツールは Java で開発されており、8 種類の XSD スキーマと変換用の設定ファイル (prefix-config.xml) を入力とし、それぞれに対応する RNG スキーマを出力する。

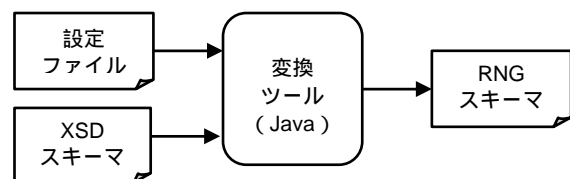


図 2 スキーマの変換

† 大阪工業大学 情報科学研究科

‡ 大阪工業大学 情報科学部 情報メディア学科

## 4 スキーマの変換

### 4.1 型定義と要素定義の変換

CoreComponentTypes の Measure と MeasureType の定義を例として図 3 に示す。XSD の要素と型は RNG ではパターンとして定義する。RNG のパターン名には名前空間を設定できないため、"ubl-バージョン-スキーマの接頭辞-要素名 (または型名)"として重複を避けた。スキーマの接頭辞は変換用の設定ファイルでスキーマごとに設定しておく。

XSD記述例	RNG記述例
<pre>&lt;xsd:element   name="Measure"   type="MeasureType" /&gt; &lt;xsd:complexType   name="MeasureType"&gt;   &lt;xsd:simpleContent&gt;   &lt;xsd:extension     base="xsd:decimal"&gt;   &lt;xsd:attribute     name="unitCode"     type="xsd:token"     use="optional"/&gt;   &lt;/xsd:extension&gt;   &lt;/xsd:simpleContent&gt; &lt;/xsd:complexType&gt;</pre>	<pre>&lt;define name=   "ubl-1.0-cct-Measure"&gt;   &lt;element     name="Measure"&gt;     &lt;ref name=       "ubl-1.0-cct-MeasureType"/&gt;   &lt;/element&gt; &lt;/define&gt; &lt;define name=   "ubl-1.0-cct-MeasureType"&gt;   &lt;data type="decimal"/&gt;   &lt;optional&gt;   &lt;attribute     name="unitCode"&gt;   &lt;data     type="token"/&gt;   &lt;/attribute&gt;   &lt;/optional&gt; &lt;/define&gt;</pre>

図 3 型定義と要素定義の変換

### 4.2 派生型定義の変換

XSD スキーマは型を派生させて新たな型を定義できる。最下位の派生元は XSD データ型 (例: decimal, string) である。派生には元の型の値や属性値を限定する制限と元の型に内容を追加する拡張がある。どちらも型を参照して新たな型を定義することになる。

RNG は拡張はできるが制限はできない。よって派生は派生元を辿り、最下位までの型情報を全て含んだ新しいパターンを定義する必要がある。

## 5 スキーマ情報の追加

### 5.1 バージョニングとカスタマイゼーション

UBL のバージョニングとカスタマイゼーションはスキーマを派生させて行う。例えば ver.1.1 のスキーマライブラリは ver.1.0 のスキーマライブラリを派生させて作られる。したがって新バージョンのスキーマライブラリは、その前のバージョンのスキーマライブラリを包括していることになる。

### 5.2 スキーマ情報

異なるバージョンがカスタマズされたスキーマで取引を行う場合、取引当事者間で共通する最新のスキーマを用いる。つまり ver.1.2 のスキーマと ver.1.1 カスタムスキーマで取引を行う場合、共通する最新バージョンである ver.1.1 のスキーマを採用するということである。これは 5.1 で述

べたように、バージョニングとカスタマイゼーションが派生元のスキーマを包括しているため可能となる。

このとき、そのスキーマがどのスキーマを派生させて作られたものかという情報が必要になる。この情報は Post Schema Validation Infoset (PSVI) に含まれ、XSD の特徴となっている。しかし RNG では PSVI の取得はできず、スキーマの派生元を辿ることはできない。そこで RNG スキーマに必要なスキーマ情報を追加した。

### 5.3 スキーマ情報の追加

追加したスキーマ情報は以下の 4 つである。

変換元になった XSD スキーマの型名 AType

AType の派生元となった XSD スキーマの型名 BType

AType の派生の種類 (拡張か制限)

BType の名前空間 (スキーマの個有値)

スキーマ情報は RNG スキーマの注釈として追加した。これにより RNG スキーマにおいても派生元スキーマ情報を取得でき、異なるバージョン間で取引を行うことができる。

スキーマ情報の例として UnspecializedDatatypes の MeasureType の RNG スキーマの一部を図 4 に示す。この例では変換元の MeasureType が CoreComponentTypes (ver.1.0) の MeasureType を制限して定義されたという情報が含まれている。

```
<define name="ubl-1.0-udt-MeasureType">
  <a:documentation>
  <xsdTypeInfo>
  <xsdType>MeasureType</xsdType>
  <xsdBaseType>MeasureType</xsdBaseType>
  <xsdDerivedType>restriction</xsdDerivedType>
  <xsdBaseTypeNs>urn:oasis:names:tc:ubl:
    CoreComponentTypes:1:0</xsdBaseTypeNs>
  </xsdTypeInfo>
</a:documentation>
```

図 4 スキーマ情報

## 6 まとめ

開発した RNG スキーマライブラリは XSD スキーマライブラリと等価な XML インスタンスを定義できるため、通常の取引においては相互運用性があると言える。しかし異なるバージョン間の取引においては、今回追加したスキーマ情報を取得するためのツールが必要となる。このツールを開発し、RNG スキーマを用いて異なるバージョン間で取引を行うことが今後の課題である。

### 参考文献

- [1] OASIS Universal Business Language TC  
[http://www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=ubl](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=ubl)
- [2] RELAX NG Specification  
<http://www.oasis-open.org/committees/relax-ng/spec-20011203.html>
- [3] 内藤広志、衣川勇樹、井戸翔子、田口華香、二宮智映、細島賢：UBL 用の Relax NG スキーマの開発  
第 66 回情報処理学会全国大会公演論文集 (2004)
- [4] OASIS  
<http://www.oasis-open.org/home/index.php>