

## 漁業センサスの電子化におけるXML標準の利用について

メタデータ	言語: Japanese 出版者: 水産大学校 公開日: 2024-10-11 キーワード: data; data collections; census; surveys; data processing 作成者: 楢取, 和明, 瓜倉, 茂, 青木, 邦匡 メールアドレス: 所属:
URL	<a href="https://fra.repo.nii.ac.jp/records/2011903">https://fra.repo.nii.ac.jp/records/2011903</a>

This work is licensed under a Creative Commons Attribution 4.0 International License.



# 漁業センサスの電子化におけるXML標準の利用について

梶取和明, 瓜倉 茂, 青木邦匡

## Applying XML standards to the Fishery Census codebook

Kazuaki Kajitori<sup>†</sup>, Shigeru Urikura and Kunimasa Aoki

**Abstract** : Social research data are often distributed with the research metadata such as research areas, definitions of research concepts, layouts of physical data, etc. The collection of these meta information for a research is called a codebook. Now that we know the usefulness of electric data of a research, why don't we computerize the codebook, too? In this paper, we discuss standards for describing a codebook, and apply DDI 3, a new XML-based candidate of the standard, to the codebook of the Fishery Census of Japan. We argue what we need to put DDI 3 (or other candidates) to practical use in social sciences.

**Key words** : Data, Data collections, Census, Surveys, Data processing

### はじめに

公刊されている社会調査データを利用するに当たっては、調査期間、調査対象・地域、調査方法、調査項目、各種調査概念の定義、データ書式、諸注意などの情報が必要である。

実際、11次漁業センサスの報告書<sup>1)</sup>の場合、「総括編」の巻の始めに「利用者のために」という章があり、その章の目次はつぎのようなものである。

#### I 漁業センサスの沿革

- 1 漁業センサス以前の基本調査
- 2 年次別漁業センサスの概要

#### II 2003年（第11次）漁業センサスの概要

- 1 我が国水産業をめぐる動きと2003年（第11次）漁業センサスの役割
- 2 2003年（第11次）漁業センサスにおける主な視点
- 3 2003年（第11次）漁業センサスの主な改正点
- 4 調査の体系
- 5 調査の定義及び約束事項
- 6 数値及び記号の表示
- 7 大海区区分図

#### 8 報告書等の刊行一覧

- 9 2003年（第11次）漁業センサスに爛する問い合わせ先

巻末には、漁業センサスの調査に使われた「調査票」が11種類載っている。

また、家計調査の電子データ<sup>3)</sup>の場合、「データファイル説明書」という冊子がデータディスクとは別についている。

調査によって得られた調査対象に関するデータは「狭義の調査データ」なので単に「調査データ」と呼び、上述のような調査に関する情報を「調査のメタデータ」と呼ぶこともある。「メタデータ」という言葉はあまり一般的ではないが、家計調査の「データファイル説明書」のように電子データに付随するものの場合、電子データに使われるコードが何を意味するのかを主に書いているという意味で「コードブック」ということもある。本論では、狭義の調査データを調査データと呼び、メタデータの代わりにコードブックと表現することにする。

我々は、調査データの電子化についてその意義はすでにいくつかの論文<sup>4,5,6)</sup>で論じている。本論ではコードブックの電子化と、調査データも含めた電子化フォーマットの

2008年9月19日受付. Received September 19, 2008.

水産大学校水産流通経営学科 (Department of Fisheries Distribution and Management)

<sup>†</sup> 別刷り請求先 (Corresponding author): kajitori@fish-u.ac.jp

標準について論ずる。

## XMLによるコードブックの電子化について

さて、調査データが電子データで得られているとして、その上でコードブックも電子化するメリットは次のようである。

- ・コードブックに記されている情報（調査項目名、地名、調査データの構造、調査項目の階層構造など）をコンピュータで利用するとき、入力の手間が省ける。したがって、入力ミスによる誤りの混入を防ぐことができる。
- ・調査データとコードブックが共に電子化されることで、コンピュータによる自動処理がしやすくなる。

たとえばデータをリレーショナルデータベースに納めて漁業センサスのデータベースを構築するには、調査データとコードブックの対応を確かめてリレーショナルデータベースを設計し、コードブックの情報を入力し、調査データを格納する必要があり、これらの作業を年次ごとにするのに大変手間がかかった<sup>4)5)7)</sup>。ここでコードブックが電子化されかつ調査データとの対応が機械処理できるように書かれていれば、この作業は大幅に軽減される。

ただし、電子化のメリットを最大限に生かすには、ただメタ情報を雑然と電子メディアに書き記せばよいわけではない。以下のような点に留意する必要がある。

- ・細かいこと、複雑なことを含め、コードブックのすべての情報を表現できなければならない。
- ・それらは、コンピュータによって効率的に処理できるように表現されていなければならない（機械可読性）。
- ・それらは、人間にも理解できるように書かれている必要がある。少なくとも、人間に理解しやすい形式に簡単に変換できるようになっていなければならない。
- ・いろいろな形式への変換性、データ流通への適格性を持たなければならない。

われわれは、こうした目的のためのデータ表現形式としてXMLとXML Schemaを提案した<sup>5)</sup>。XML<sup>8)</sup>はタグセットを定義して文書表現するものであるが、XMLのタグセットを定義するやりかたとしてはXML Schema<sup>9)</sup>が代表的である。XML Schemaは、必要なタグセットの仕様をXML形

式で書くための定義言語であり、今後広く標準として使われると見られ、上のような条件に沿ったものといえる。

例えば、XML Schemaの例として、〈QuestionScheme〉というタグ（XML Schemaでは要素と呼ぶ）を〈Question〉というタグと共につぎのように定義する。ここで、〈QuestionScheme〉は調査票を表すためのタグ、〈Question〉は質問を表すためのタグである。

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
<xsd:element name="QuestionScheme" type="QuestionSchemeType"/>
<xsd:complexType name="QuestionSchemeType">
<xsd:sequence maxOccurs="unbounded">
<xsd:element ref="Question"/>
</xsd:sequence>
<xsd:attribute name="label" type="xsd:string" use="required"/>
</xsd:complexType>
<xsd:element name="Question" type="xsd:string"/>
</xsd:schema>
```

2行目で、QuestionScheme要素を定義して、その型QuestionSchemeTypeを3行目から8行目までで定義している。9行目で、Question要素を定義している。

ここで定義したタグを使って、個人漁業経営体への単純なアンケートとして、市区町村、漁業地区、世帯主氏名を問う調査票を

```
<?xml version="1.0"?>
<QuestionScheme label="個人漁業経営体調査票">
<Question>市区町村</Question>
<Question>漁業地区</Question>
<Question>世帯主氏名</Question>
</QuestionScheme>
```

と表すことができる。

しかし、XML Schemaで社会調査データ（コードブック）を表現するタグセットを定義する仕方にも標準があったほうがよい。それは、以下の理由による。

- ・社会調査データ（コードブック）表現の標準として広く使われることによって、その形式を他のユーザニーズに合わせた形式（Excel、リレーショナルデータベース、報告書、利用説明書などなど）に変換するツール類が整備されやすい。
- ・その他、社会調査データ（コードブック）表現の標準として広く使われることによって、その形式をサポートするサービスが生まれやすくなる。

いろいろな形式への変換は、XMLの機械可読性がよく発揮される場所であって、XMLにはXSLTといわれる効率的な変換方法が用意されている。たとえば、上のアンケート票を表すXML文書を、つぎのXSLTスクリプトで処理すると、

```
<?xml version="1.0" encoding="EUC-JP"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="QuestionScheme">
<html><body><table border="1">
<tr>
<th align="center">
<xsl:attribute name="colspan">
<xsl:value-of select="count(descendant-or-self::*[count(descendant::*)=0])"/>
</xsl:attribute>
<xsl:value-of select="@label"/>
</th>
</tr>
<tr>
<xsl:apply-templates/>
</tr>
<tr>
<td></td><td></td><td></td>
</tr>
</table></body></html>
</xsl:template>
<xsl:template match="Question">
<td>
<xsl:value-of select="."/>
</td>
</xsl:template>
</xsl:stylesheet>
```

つぎのような、HTMLの表に変換できる。

```
<html><body><table border="1">
<tr><th align="center" colspan="3">個人経営体調査票</th></tr>
<tr>
<td>市区町村</td>
<td>漁業地区</td>
```

```
<td>世帯主氏名</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</table></body></html>
```

これはウェブブラウザで見れば、

Table 1 . An example of XSLT transformations.

個人経営体調査票		
市区町村	漁業地区	世帯主氏名

のように見える。XSLTは手続き的であるより宣言的な言語であるので簡略に書けるし、より複雑なXML文書にも対応している。

社会調査データを表現するための標準が存在し、その標準によって漁業センサスデータがコードブックを含め電子化データとして表されているとすると、そのデータの有用性は飛躍的に高まる。たとえば、他の標準的なデータ形式であるExcel、リレーショナル（SQL）などへの変換とか、市区町村、都道府県庁レベルでの集計とかがかなりの部分自動化できるであろう。より進んでたとえば、漁業センサス報告書用の各種集計表Excelシートを自動生成するようなツールを組むこともできるだろう。また、社会調査データを広く有効利用するための流通フォーマットとして認識されていこう。

このような社会調査データを表現するフォーマットの標準としては、DDI<sup>10)</sup>、SDMX<sup>11)</sup>、ISO11179<sup>12)</sup> などがある。このうち、個票データをコーディングすることに適しているのはDDIであり、DDIの最新バージョンDDI3.013) は、関連調査の比較などにも注意を払っている。DDI3.0の仕様は、上記のXML Schemaの形式で書かれており、DDI3.0で定義されたタグを使ってコードブックをXML文書として表すようになっている。DDI（Data Documentation Initiative）は、DDI 1が2000年にリリースされ、DDI 2が2003年にリリースされ、DDI 3は2008年になってリリースされた。

漁業センサスデータは、元データが何種類かの調査票データの集合体であるから、個票データである。以前に述べた<sup>6)</sup> ように、データは個票ベースで管理するのがもっ

とも望ましいので、次章では漁業センサスのコードブックをDDI3.0でコーディングすることを検討する。

## DDI 3 による漁業センサスコードブックのコーディング

漁業センサスのコードブックをDDI3.0で書くやりかたは一通りではない。漁業センサスは、海面漁業調査、内水面漁業調査、流通加工調査の三つに分かれている。以下は、多年次の漁業センサスコードブックをその違いを比較できるようにDDI3.0を使って階層構造に書いた例を、タグによらず普通の用語で表したものである。海面漁業経営体調査以外の調査については下位階層は省略している。

### 漁業センサス

漁業センサスの概要・目的

漁業センサスに現れる概念

### 海面漁業調査

漁業従事者世帯調査の調査方法

海面漁業経営体調査のデータ表現

### 海面漁業経営体調査

海面漁業経営体調査の変数

個人・組合・共同経営調査

個人・組合・共同経営調査の質問事項

6次調査

～

11次調査

会社・官公庁・学校・試験所調査

会社・官公庁・学校・試験所調査の質問事項

6次調査

～

11次調査

漁業従事者世帯調査

(以下の階層省略)

漁業管理組織調査

(以下の階層省略)

海面漁業地域調査

(以下の階層省略)

内水面漁業調査

(以下の階層省略)

流通加工調査

(以下の階層省略)

DDI3.0によるこの階層構造には、オブジェクト指向と呼ばれる以下のような機能が組み込まれている。

- ・より上の階層で定義された情報はより下の階層でも自動的に使うことができる。これは継承と呼ばれる。これにより上の階層で、各調査、各年次に共通することがらを一括して表すことができる。
- ・上の階層で定義された情報をより下の階層で書き換える又は書き足すこともできる。

これにより、各調査間、各年次間での違いを表すことができる。たとえば、上の階層構造の2行目は漁業センサス全体の概要・目的であるが、下の階層でその調査独自の概要・目的を書くことができる。XMLコードの例は、以下で取り上げる。

DDI3.0は、これらの階層構造とそのディテールを表すためのタグを定義する以下のようなXML Schemaファイル群からなる。

Table 2 . DDI3.0 schema files.

ファイル名
archive.xsd
comparative.xsd
conceptualcomponent.xsd
datacollection.xsd
dataset.xsd
dcelements.xsd
ddi-xhtml11-model-1.xsd
ddi-xhtml11-modules-1.xsd
ddi-xhtml11.xsd
ddiprofile.xsd
group.xsd
instance.xsd
logicalproduct.xsd
physicaldataprotuct.xsd
physicaldataprotuct_ncube_inline.xsd
physicaldataprotuct_ncube_normal.xsd
physicaldataprotuct_ncube_tabular.xsd
physicaldataprotuct_proprietary.xsd
physicalinstance.xsd
reusable.xsd
simpledc20021212.xsd
studyunit.xsd
xml.xsd

各ファイルはモジュールと呼ばれるDDI3.0における機能のまとまりに対応している。例えば、instance.xsdはDDI3.0で書かれるコードブックの一番外側の枠の部分のタグを定義している。studyunit.xsdは、最小の調査単位を表すためのモジュールで、上の構造では、各年次調査を記述する。group.xsdは、上のような階層構造で継承・上書きをするために使われる。logicalproduct.xsdは、変数の定義をする

ためのモジュールである。conceptualcomponent.xsdは、調査の諸概念（地理データを含む）を記述するためのモジュールである。physicaldataprotuct.xsdは、調査データのフォーマットを定めるモジュールである。physicalinstance.xsdは、調査データファイルについての記載をするためのモジュールである。comparative.xsdは、group以外の方法で直接比較をするためのモジュールである。resusable.xsdは、いろいろなモジュールで使われるタグをまとめたものである。

その他、datacollection.xsdは、アンケートの内容、実施方法を記すためのモジュールである。ncubeがついたモジュールは、クロス集計表を記述するためのモジュールである。

我々が上の階層構造をDDI3.0でコーディングしたとき使ったモジュールの説明を簡単に述べたが、DDI3.0が漁業センサスのコードブックを記述するのに少なくとも骨格的には十分な機能（タグセット）を持っていることはお分かりいただけるであろう。以下に、DDI3.0のタグを使って上の階層構造を表してみよう。DDI3で本来必須とされるものを含め詳細は省略してある。

```
<?xml version="1.0"?>
<ddi:DDIInstance>
  <g:Group id="gyosGroup">
    <r:Citation>
    </r:Citation>
    <g:Abstact>
    </g:Abstact>
    <g:Purpose>
    </g:Purpose>
    <r:SeriesStatement>
    </r:SeriesStatement>
    <g:Concepts>
      <c:ConceptualComponent>
      </c:ConceptualComponent>
    </g:Concepts>
    <g:SubGroup id="kaimen_Group">
      <g:SubGroup id="kaimen_keieitai_Group">
        <g>DataCollection>
        </g>DataCollection>
        <g:LogicalProduct>
        </g:LogicalProduct>
        <g:PhysicalDataProduct>
```

```
</g:PhysicalDataProduct>
    <g:SubGroup id="kojin_kumiai_kyodo_Group">
      <g:StudyUnit>
        <s:StudyUnit id="gyos06_kojin_kumiai_kyodo">
        </s:StudyUnit>
      </g:StudyUnit>
      ...
      <g:StudyUnit>
        <s:StudyUnit id="gyos11_kojin_kumiai_kyodo">
        </s:StudyUnit>
      </g:StudyUnit>
    </g:SubGroup>
    <g:SubGroup id="kaisha_kankocho_gakko_shikenjo_Group">
    </g:SubGroup>
  </g:SubGroup>
  <g:SubGroup id="naisuimen_Group">
  </g:SubGroup>
  <g:SubGroup id="ryutsukako_Group">
  </g:SubGroup>
</ddi:DDIInstance>

  上記階層構造は、DDI3.0では、Group, SubGroupの入れ子構造で表現されている。

  ここで、上記のXML階層構造のうち13, 14行目、全体に関わる調査概念（concept）を述べるところ、

  <c:ConceptualComponent>
  </c:ConceptualComponent>

  の部分の詳細の一部を以下に示す。ここでは漁業センサス全体に関わる概念のほかに各調査の基本概念も全体への見通しのために書いているのであるが、全部書くと長いので一部に止めるのである。内容は、11次漁業センサス報告書の「利用者のために」からのものである。

  <c:ConceptualComponent>
  <c:ConceptScheme id="gyos_ConceptScheme">
  <c:Concept id="gyos_Concept_1">
```

```

<r:Name>過去1年間</r:Name>
<r:Label>Kakoichinenkan</r:Label>
<r:Description>平成14年11月1日から平成15年10月31日までの期間</r:Description>
</c:Concept>
<c:Concept id="gyos_Concept_2">
<r:Name>漁業経営体</r:Name>
<r:Label>GyogyoKeieitai</r:Label>
<r:Description>過去1年間(H14.11.1~H15.10.31)に利潤又は生活の資を得るために、生産物を販売することを目的として、海面において水産動植物の採捕又は養殖の事業を行った世帯又は事業所をいう。</r:Description>
</c:Concept>
<c:Concept id="gyos_Concept_3">
<r:Name>漁業経営体ID</r:Name>
<r:Label>GyogyoKeieitaiID</r:Label>
<r:Description>多年度にわたって漁業経営体を同定するためのID</r:Description>
</c:Concept>
<c:Concept id="gyos_Concept_4">
<r:Name>漁業地区</r:Name>
<r:Label>GyogyoChiku</r:Label>
<r:Description>市区町村の区域内において、共通の漁業条件の下に漁業が行われる地区として、共同漁業権を中心とした地先漁業の利用など漁業に係る社会活動の共通性に基づいて農林水産大臣が設定するものをいう。</r:Description>
</c:Concept>
<c:Concept id="gyos_Concept_5">
<r:Name>内水面漁業</r:Name>
<r:Label>NaisuimenGyogyo</r:Label>
<r:Description>共同漁業権の存する天然の湖沼その他の湖沼で農林水産大臣が定める湖沼において水産動植物の採捕の事業又は内水面（浜名湖、中海、猿澗湖、風蓮湖及び厚岸湖は除く。以下同じ）において営む養殖業をいう。</r:Description>
</c:Concept>
<c:Concept id="gyos_Concept_6">
<r:Name>経営組織</r:Name>
<r:Label>KeieiSoshiki</r:Label>
<r:Description>漁業経営体あるいは内水面漁業経営体を経営形態別に分類する区分をいう。個人経営体、会社、漁業

```

```

共同組合、漁業生産組合、共同経営、官公庁・学校・試験場に分類される。
</r:Description>
</c:Concept>
<c:Concept id="gyos_Concept_7">
<r:Name>魚市場</r:Name>
<r:Label>Uoichiba</r:Label>
<r:Description>過去1年間に漁船により水産物の直接水揚げがあった市場直接水揚げがなくても、陸送により生産地から水産物の搬入を受けて、第1次段階の取引を行った市場をいう。
</r:Description>
</c:Concept>
<c:Concept id="gyos_Concept_8">
<r:Name>水産加工場</r:Name>
<r:Label>SuisanKakojo</r:Label>
<r:Description>販売を目的として過去1年間に水産動植物を他から購入して加工製造を行った事業所及び原料が自家生産物であっても加工製造するための作業場又は工場と認められるものを有し、その製造活動に専従の従業者を使用し、加工製造を行った事業所をいう。
</r:Description>
</c:Concept>
</c:ConceptScheme>
</c:ConceptualComponent>

```

8個のconceptを記述するだけで、60行以上も費やしている。これ以外に、地理情報もConceptulaComponentのところ別に書くことができるようになっている。一般に、以下のようないくつかの理由で、XML文書はかなり冗長になる。

- ・各要素に開始タグと終了タグがある。同様のデータが多数あると、すべて同じ開始・終了タグではさまなければならない。
- ・機械処理のために表現が厳密になり省略ができない。
- ・汎用性（いろいろな社会調査データを表現できるように）を確保するためSchemaの構造が複雑になり多くのタグが必要になる。

XML文書は人間がすべて直接入力するには冗長過ぎる。かろうじて人間が直接読める程度のものである。XMLの入力に当たっては入力支援機能が必要であるし、人間が読

むに当たっても下位階層を隠したり表示したりする階層構造を伸縮する機能が必要である。well-formedness (XMLの書き方として正しいこと), validity (XML Schemaファイルに矛盾しないこと) の検査機能も必要である。実際, XML文書を効率的に読み書きするために, こうした機能を備えたXMLエディタなるアプリケーションがいくつもある。

今回, 漁業センサスコードブックのDDI3 コーディングに当たっては, 普段使い慣れているテキストエディタ (vi) とともにEditiX<sup>14)</sup> なるXMLエディタのFree Versionを使用した。

しかしながら, 汎用のXMLエディタだけでは不十分で, DDI3.0用の編集ツールが望まれる。それは, table 2 のようなDDI3.0のスキーマファイルを読んでコードブックをDDI3 コーディングするのは, XML, XML Schemaについての知識があっても大変だからである。

対話的なインターフェイスで, DDI3 の機能を駆使したコードブックのコーディングができるような編集ツールがないと広くDDI3 が使われるのは難しいだろう。

この章の最後に, DDI3 における調査間の比較の方法を例示しておこう。

11次漁業センサスにおいては, 最盛期の海上作業人数を男女別に計上している (10次以前は, 男女計のみ)。海面漁業経営体調査全体のSubGroup (上のDDI3 階層の例で言うと, kaimen\_keieitai\_Group) で, 最盛期の海上作業従事者人数の変数 (var 1) を次のように定義してあるとする。

```
<l:Variable id="var 1">
```

変数の内容

```
</l:Variable>
```

それに対して, 11次の漁業経営体調査 (個人・組合・共同経営) のSubGroupで, var 1 を削除 (Delete) し, 新しく男女別の変数var 1\_1, var 1\_2 を加える (Add)。

```
<l:Variable id="var 1" action="Delete">
```

変数の内容

```
</l:Variable>
```

```
<l:Variable id="var 1_1" action="Add">
```

変数の内容

```
</l:Variable>
```

```
<l:Variable id="var 1_2" action="Add">
```

変数の内容

```
</l:Variable>
```

その上で, 次のような比較方法を海面漁業経営体調査全体のSubGroupに書いておく。

```
<cm:Comparison>
```

```
<cm:VariableMap>
```

```
<cm:SourceSchemeReference>
```

```
<r:ID>kojin_kumiai_kyodo_VariableScheme</r:ID>
```

```
</cm:SourceSchemeReference>
```

```
<cm:TargetSchemeReference>
```

```
<r:ID>kojin_kumiai_kyodo_11_VariableScheme</r:ID>
```

```
</cm:TargetSchemeReference>
```

```
<cm:Correspondence>
```

```
<cm:Commonality> 下記の相違以外は同一。
```

下記については,  $var 1 = var 1_1 + var 1_2$  の関係がある。

```
</cm:Commonality>
```

```
<cm:Difference>11次においては, 女性の労働状況や漁業関連産業における就業状況を把握するため, 漁業経営体における漁業の海上作業に従事した人が最も多かった時期の人数 (var 1) を男女別 (var 1_1, var 1_2) に把握する項目を設定。
```

```
</cm:Difference>
```

```
</cm:Correspondence>
```

```
</cm:VariableMap>
```

```
</cm:Comparison>
```

漁業センサスコードブックの一部 (漁業センサス-海面漁業調査-海面漁業経営体調査-個人組合共同経営体調査までの部分) をDDI3.0で書いてみた結果を, DDI3.0スキーマファイルのもとでvalidator (Perl XML::LibXML::Schema->validate) にかけての結果は"validated successfully"であった。

## 利用への向けての課題

前章のように実際にDDI3 コーディングをしてみるとDDI3 3について見えてくるのは次のようなことである。

- DDI3 は, 漁業センサスのコードブックをコーディングするのにほぼ十分な仕様を持っている。アンケート票の質問ごとの記入の仕方の指示を書く方法とか, 系列でない調査の比較方法とか, 機械処理を視野にさらに詰めて

いく部分が残されている。実際に使うアンケート用紙のレイアウトなどは、無理にDDIで書くより、外部のレイアウトを別に用意してそこに流し込む内容をDDIで記述の方がよいであろう。

- ・人間にとって意味が取りやすいXMLがベースになっているとはいえ、社会調査全般における汎用性を目指したDDI 3は抽象的、かつ複雑である。また、DDI 3のタグ名はすべて英語で日本人にはピンとこないし、馴染みにくい。編集段階をサポートするツールがないと、事実上使用できない（コードブックをDDIコーディングする人間が出てこない）。
- ・DDI 2用にはNesstar publisherという専用エディタ<sup>15)</sup>がある。しかし、より抽象度の高いDDI 3用にはいっそう望まれるもののまだ専用エディタは開発初期<sup>16)</sup>の段階である。DDI 3.0が2008年の4月に公式に出たばかりで、支援ツール類<sup>16)</sup>はこれから揃ってくるであろう。すでにDDI 3.0の使用例<sup>16)</sup>の中には、(XML表現した) Excel ファイルをXSLTにかけて作り出したものが見られるが、こういったノウハウも広まっていく必要がある。
- ・DDI 3の意図した使用法をXML Schemaファイルのみで規定するのは無理がある。User Guide文書およびHTMLで書かれたDDIモジュールの解説ページがあるので、これらでかなり分かるが、やはりDDI 3の使用法を理解させるには解説とともに様々な使用例があった方がよい。これらサンプルは存在する<sup>16)</sup>が、まだまだ少ない。
- ・社会調査データという比較的狭い領域で、XML, XML Schema, DDIの知識を身につけた人材を育成できるか。社会全体を考えても数多くのニッチ領域ごとに専門知識のある人材を育成できるか。
- ・社会調査データという比較的狭い領域の標準に対して、どれほどのサポートが得られるか。社会全体を考えてもニッチごとに標準が立ってしまうとサポートが追いつかないのではないか。

XMLは、1998年にWorld Wide Web Consortiumによってその1.0（現在は1.1）が勧告されて以来、すでに広くデータ記述言語として使われてきており、XMLの汎用技術も

ソフトウェアやノウハウとして蓄積されてきている。XML SchemaはXMLのスキーマ言語としてまだ比較的新しいが、XML汎用技術を使って固有のサポート技術を蓄積していくものと思われる。DDI 3は出てまだ間もないが、上に記したような懸念はあるものの、XMLのインフラに支えられて今後広く使われていくことを期待したい。

## 参考文献

- 1) 農林水産省統計部, 2003年(第11次) 漁業センサス報告書, 2007.
- 2) 農林水産省統計部, 2003年(第11次) 漁業センサス海面漁業に関する電子統計書, 2007.
- 3) 統計情報研究開発センター, 平成18年家計調査年報集計結果データ, 2007.
- 4) 楫取和明, 井元康裕, 三輪千年: 漁業センサスをはじめとする水産関連電子データの利用方法に関する一考察. 農林統計調査, 50(8), 19-32 (2000).
- 5) 井元康裕, 楫取和明, 三輪千年: 漁業情報システム化推進に向けて. 漁業情報利用, 第1号, 6-15 (2003)
- 6) 楫取和明: 水産関連電子データの利用手法について. 釜慶大学校 学術交流懇談会 講演論文集 (IV), 83-92 (2004).
- 7) 楫取和明, 瓜倉 茂, 青木邦匡: 漁業センサスデータベースの更新について. 水大校研究報告, 56, 349-354 (2007).
- 8) Extensible Markup Language (XML), <http://www.w3.org/XML/>.
- 9) XML Schema, <http://www.w3.org/XML/Schema/>.
- 10) <http://www.ddialliance.org/>.
- 11) SDMX, <http://http://www.sdmx.org/>.
- 12) ISO11179, <http://metadata-standards.org/11179/>.
- 13) <http://www.ddialliance.org/ddi3/index.html>.
- 14) <http://www.editix.com/>.
- 15) <http://www.nesstar.com/>.
- 16) <http://www.ddialliance.org/DDI/ddi3/use-cases.html>.