

文書記述言語の教育の課題 — 第24回VMA研究会のテーマ設定によせて

Issues for education of document processing and description languages – Introduction to the topic of VMA-24

小町 祐史*
Yushi KOMACHI

大野 邦夫**
Kunio OHNO

* 大阪工業大学情報科学部, Osaka Institute of Technology

** 職業能力開発総合大学校, Polytechnic University

1. 背景

VMA研究会は、画像の入力・合成・加工・処理、それらの応用、およびそれらの結果として生じるVisual Applianceについて論じると共に、画像電子学会が今後扱うべき新しいトピックを見出し、そこでの議論の立ち上がり応じて新研究会の設立等を図ることを主眼として活動を行っている。

文書記述言語については、これまでISO/IEC JTC1/SC34, W3Cなどの活動と同期して関連規格の規定内容に関する議論を展開し、規格制定に寄与してきた。2005年からは、Conference“XML開発者の日”を主催[1][2][3]して、文書記述言語関連の実装技術に関する情報交換の場を提供してきた。

SGMLの構文に基づいたHTMLはWeb文書の記述に利用されてWebの普及を促進し、Web環境でのHTMLの手軽さを継承したままSGMLの一般化(generalize)を可能にしたXMLは、構造化されたデータを記述するさまざまなアプリケーションに利用され、大学での情報技術の教育の対象として取り上げられつつある。

そこでVMA研究会を運営するVMA研究委員会のコアメンバ(小町, 大野, 他)は、文書記述言語の規定内容と実装だけでなく、文書記述言語の教育についても検討の場を提供することが必要であろうと考え、“文書記述言語の教育”というテーマ設定を行って講演募集を行った。

その結果、他大学に先駆けて学部学生に対するXML教育の専門科目を開講している大阪工業大学情報科学部において、“構造化文書処理”[4]を2002年度から担当している内藤広志先生から講演をいただけることになった。

日本ユニテックの奥井康弘氏にも講演のご快諾をいただいた。奥井氏は、文書記述言語の国際標準化を担当するISO/IEC JTC1/SC34およびその前身のJTC1/SC18/WG8のエキスパートとして国際規格原案作成に寄与され、XML関連技術の本格的解説書をはじめ国内で出版されて、XML教育の先駆者として現在も活躍されている。

この両名のご講演があれば、VMA研究会としてはじめての“文書記述言語の教育”に関する研究会開催が可能と判断し、第24回VMA研究会(文書記述言語の教育)の開催を決定した。

2. 文書記述言語の教育に関する課題

VMA研究委員会のコアメンバによる議論に際して、コアメンバの念頭にあった文書記述言語の教育に関する課題(主として大学の学部学生を対象とするときの課題)は次のとおりである。

- (1) 文書構造記述の言語
 - HTML, XML, スキーマ言語

- HTMLの構文としてSGML, DTDを扱うか
 - スキーマ言語はDTDだけか, XML構文, コンパクト構文まで含めるか
- (2) 構造変換およびスタイル指定の言語
- HTMLのデフォルトスタイルとXSLTによるHTMLへの構造変換
 - CSS (背景となる組版に関する解説が必要)
 - XSL-FO (背景となる組版に関する詳細解説が必要)
 - XSLTで用いるX-Path
- (3) API
- (4) XML応用言語
- (5) 教材
- (6) 演習環境

3. 各課題に対する回答事例

前節に示した各課題については, 内藤先生と奥井氏とが, 長年のXML教育経験に基づいて回答事例を示して下さることが期待される。

VMA研究委員会のコアメンバの小町, 大野も幾つかの回答事例を用意している。大野は, “XMLとS式による情報記述と把握理解の比較”と題して小文をまとめた。小町は, 2008年度から大阪工業大学情報科学部の“構造化文書処理”[5]を担当し, 内藤先生とは異なる学科の学生に対して幾分異なるシラバスに基づいて講義を行なっている。その概要を付録Aに示す。

4. 議論の展開

第24回VMA研究会に提出された講演論文に関する議論をきっかけとして, 今後我が国での文書記述言語関連技術の教育がより一層充実することを期待する。

文献

- [1] 第8回XML開発者の日, <http://www.hi-ho.ne.jp/y-komachi/committees/vma/confs/xml-dev/xml-dev-8.htm>
- [2] 第9回XML開発者の日, <http://www.hi-ho.ne.jp/y-komachi/committees/vma/confs/xml-dev/xml-dev-9.htm>
- [3] 第10回XML開発者の日, <http://www.hi-ho.ne.jp/y-komachi/committees/vma/confs/xml-dev/xml-dev-10.htm>
- [4] <http://www05.ofc.oit.ac.jp/syllabus/syllabus/search/SyllabusInfo.do?nendo=2009&kogikey=1C25702A0>
- [5] <http://www05.ofc.oit.ac.jp/syllabus/syllabus/search/SyllabusInfo.do?nendo=2009&kogikey=1Q25702A0>

付録A 小町が担当する文書記述言語関連講義

A.1 講義概要

小町が担当する文書記述言語関連講義の概要を示すため, 大阪工業大学シラバスの抜粋(科目:構造化文書処理, 担当:小町)を次に掲載する。

科目名 (英文名)	構造化文書処理 Structured Document Processing
学科	情報科学部 コンピュータ科 (情報科) 学科
年次	3年次
期間	後期

授業のねらい・概要	文字列と補助情報から成る文書情報に構造を与え、論理構造とスタイル指定とを分離した電子化文書の扱いを習得する。特にXMLとそのスキーマ言語であるRELAX NGとに着目して、その文法を習得するとともに、それらを用いた文書情報記述を可能にする。さらにXML文書の可視化のためのXSL (XSLTおよびXSL-FO) による構造変換とスタイル指定にも言及する。		
授業計画	回数	テーマ	内容・方法等
	第1回	文書の構造	文字列と補助情報から成る文書情報に構造を与える。
	第2回	文書情報要素としての文字	文書情報の基本構成要素としての符号化文字の扱いを知る。
	第3回	HTMLによる構造記述	文書処理モデルに基づく最もシンプルな例としてHTML文書を取りあげ、その構文と特徴を知る。
	第4回	CSSによるスタイル指定	HTML文書に対する可視化への指示としてのCSS記述に着目して、CSSの構文を概観すると共に、文書構造とスタイル指定との分離の意義を知る。
	第5回	文書のスタイルオブジェクト	文書構造を適切に表現するための各種のスタイルオブジェクトを知る。
	第6回	SGMLからXMLへ	XMLの背景としてSGMLを知り、その拡張としてのXMLの位置付けを理解する。XMLの文法の詳細も習得する。
	第7回	HTMLからXHTMLへ	SGMLからXMLへの移行の具体例としてHTMLからXHTMLへの移行を知る。
	第8回	DTDによる構造記述	DTDによる構造記述を習得する。
	第9回	XML名前空間とRELAX NG (XML構文)による構造記述	XML名前空間の意義と構文を習得する。スキーマ言語としてRELAX NGを取りあげ、そのXML構文を知る。
	第10回	RELAX NG (簡潔構文)による構造記述	RELAX NG (簡潔構文)を知り、スキーマモジュールの構成を習得する。
	第11回	XSLT (1)	XSLTの文法を習得する。
	第12回	XSLT (2)	XSLT指定を用いて、XML文書をHTMLブラウザで可視化する。
	第13回	XSL-FO	XSL-FOによる高度なスタイル指定を行うための導入を習得する。
第14回	演習	演習によって構造化文書処理理解の要点を確認する。	
到達目標	(a) XMLの文法に従って、XML文書を書くことができる。 (b) RELAX NGの文法を理解し、XML文書の構造を記述することができる。 (c) XSLTの文法を理解し、XSLT指定を書いてXML文書をHTMLブラウザで表示できる。 (d) XSL-FOの概要を理解している。		
評価方法	授業中にレポートの提出を求めると共に、期末試験を実施する。 授業中のレポートの評価と期末試験結果とを総合的に評価する。		
成績評価基準	5: 到達目標 (a) (b) を達成し、期末試験とレポートの合計得点が80%以上。 4: 到達目標 (a) (b) を達成し、期末試験とレポートの合計得点が70%以上80%未満。 3: 到達目標 (a) (b) を達成し、期末試験とレポートの合計得点が60%以上70%未満。 2: 到達目標 (a) (b) を達成し、期末試験とレポートの合計得点が30%以上60%未満。 又は到達目標 (a) (b) を未達成、且つ合計得点が30%以上。		

	1: 期末試験・演習の合計点が30%未満。				
教材	印	教材の別	書名	著者名	出版社名
	1	◎ 教科書	XML入門教室	山田祥寛	翔泳社
	備考：必要に応じて補助教材をWeb等で配布する。				
受講心得	Web文書に興味をもち、あるていどHTMLに対する知識と関心があることが望ましい。				

A.2 講義の進め方

この授業は演習科目とはしない。しかしHTML, XMLなどは広く普及していて、Web技術の要素技術であり、情報技術者としてそれらのある程度使いこなすことも望まれている。そこでHTML, XMLなどを実際に使ってみることが必須であり、そのための教材として教科書を指定した。この教科書の内容は、広く普及しているWindows環境でこれらの記述言語の習得を目指すものであり、独習に向いている。

講義では、その内容把握に必要な参考文献を指定し、その重要箇所をWeb上に指示する。また講義で話した内容または黒板に記した内容の主要部分を講義メモとしてWebに掲載する。必要に応じて、指定教科書の指定箇所の学修を指示する。

評価は講義の中で提出する課題に関するレポートと期末試験の成果とに基づく。

A.3 講義で指定する参考文献

- [a1] 小町: 特集「電子出版・電子新聞」- 紙から電子へ, 情報処理, Vol. 39, No. 6, 1998-06
- [a2] 符号化文字関連JIS
- [a3] TR X 0033:2002, ハイパテキストマーク付け言語(HTML) 4.0, 2002-09
- [a4] JIS X 4168:2004, 段階スタイルシート 水準1(CSS1), 2004-06
- [a5] JIS X 4158:2005, XML名前空間, 2005-10
- [a6] JIS X 4177:2005, RELAX NG, XML Syntax, 2005-07
- [a7] JIS X 4177:2007, RELAX NG 追補1, Compact Syntax, 2007-09
- [a8] J. Clark, J. Cown and M. Murata, RELAX NG Compact Syntax Tutorial, OASIS, 2004-03
- [a9] JIS X 4159:2005, 拡張可能なマーク付け言語(XML)1.0, 2005-03

参考文献[a8]については、日本語訳がないため、小町による翻訳版を提示した。その一部を次に示す。

8. 順序なし内容モデル

子要素の出現順序を指定しない場合には、interleave結合子&を使う。

(1) 例

次のパターンでは、card要素の中にname要素とemail要素とが、どんな順序でも現れ得る。

```

element addressBook {
  element card {
    element name { text }
    & element email { text }
  }*
}

```

9. モジュール化

9.1 外部パターンの参照

externalキーワードの後に、外部パターンを含むファイルのURLを指定する引用符付きストリングをおく。

(1) 例: 被参照ファイルinline.rnc

HTMLのインライン要素にマッチするようなRELAX NGパターン。

```

start = inline
inline =
  (text
  | element code { inline }

```

```
| element em { inline }
# etc
)*
```

(2) 外部パターンの参照

externalを使って、(1)で定義されたパターンを参照する。

```
element addressBook {
  element card {
    element name { text },
    element email { text },
    element note { external "inline.rnc" }?
  }*
}
```

(3) 別の外部パターンの参照例

pattern1.rngとpattern2.rngとにRELAX NGパターンが書かれている。次のようにして、どちらかにマッチすればよいというパターンが書ける。

```
external "pattern1.rnc" | external "pattern2.rnc"
```

9.2 パターンの結合

同じ名前前で定義された複数のパターンを、どのように結合して一つにするかは、= の代わりに |= または &= を用いて指定できる

(1) 例: choice

```
inline.class |= element bold { inline }
inline.class |= element italic { inline }
```

これは、次のように解釈される。

```
inline.class =
  element bold { inline }
  | element italic { inline }
```

(2) 例: interleave

```
start =
  element addressBook {
    element card { card.attlist }*
  }
card.attlist &= attribute name { text }
card.attlist &= attribute email { text }
```

これは、次のように解釈される。

```
start =
  element addressBook {
    element card { card.attlist }*
  }
card.attlist =
  attribute name { text }
  & attribute email { text }
```

これは、次とも等価である。属性については、&による結合と、|による結合とは同じであるため。

```
start =
  element addressBook {
    element card { card.attlist }*
  }
card.attlist =
  attribute name { text },
  attribute email { text }
```

9.3 文法のマージ

include命令の後に、外部文法を含むファイルのURLを指定する引用符付きストリングをおく。参照された文法の中の定義は、include命令を含む文法の中に含まれる。|= または &= を用いたパターンの結合は効果的に使

える。

ファイルinline.rncに、次のinlineパターンが含まれるとする。

```
inline = inline.class*
inline.class =
  text
  | element bold { inline }
  | element italic { inline }
```

別のRELAX NGパターンは、次のようにしてこのinline.rncを利用して、それにinline要素としてcodeとemを追加できる。

```
include "inline.rnc"

start =
  element doc {
    element p { inline }*
  }
inline.class |=
  element code { inline }
  | element em { inline }
```

これは、次と等価である。

```
inline = inline.class*
inline.class =
  text
  | element bold { inline }
  | element italic { inline }

start =
  element doc {
    element p { inline }*
  }
inline.class |=
  element code { inline }
  | element em { inline }
```

さらにこれは、次と等価である。

```
inline = inline.class*
inline.class =
  text
  | element bold { inline }
  | element italic { inline }
  | element code { inline }
  | element em { inline }

start =
  element doc {
    element p { inline }*
  }
```