

XML と XSLT の初歩の学習について

頼野 温美

目 次

1	はじめに	1
1	1 論文のテーマと学習対象者	1
2	2 SGML・HTML との相違点と比較	2
3	3 XML と XSLT の現状と基礎を学習する利点	3
4	4 類似サイト・オンライン教材の有無と比較考察	4
5	5 オンライン教材にする利点	6
2	2 教材の作成まで	7
1	1 作成時に必要とした技術や知識	7
2	2 教材の構想	8
3	3 教材の構成	8
3	3 教材の作成内容	10
1	1 XML 編	10
2	2 XSLT 編	14
4	4 作成過程での問題点と工夫した点	16
1	1 問題点	16
2	2 工夫した点	18
5	5 おわりに	19
1	1 アンケートによる他者評価	19
2	2 今後の改善点	20
3	3 自己評価	23

1 はじめに

(1) 論文のテーマと学習対象者

福田ゼミのテーマは「人の役に立つものを作る」である。私がこれまで学んできた上で、必要である、あるいはあったら便利だと感じたものを実際に作ろうと思った。私がこの XML と XSLT について研究しようと思った理由は、ゼミにおいて学んだ XML を十分に理解できなかった事が大きな理由である。自分自身、XML や XSLT とは何であるかを理解出来ていなかったため、自分にも分かりやすくまた他の初心者にも分かりやすくというテーマを掲げて教材作成に取り組んだ。その他の理由としては、HTML と違いタグを自分で自由に設定できる点、また、XSLT スタイルシートを使用することで HTML 形式や TEXT 形式など様々な形式に変換できる点という XML の特徴に興味を持ったからである。

現在出版されている多くの XML 入門書は、学習内容が幅広く、膨大な情報量を要しているため初めて XML を学ぶ人間にとっては理解に苦しむ所がある。また、書籍を見ながらテキストエディタに文字を打ち込んでいかなければならないため、書籍のページを繰りながら文字を打つことが面倒くさくなり途中で飽きてしまう。その上打ち込む文字数が多いので、初心者にとっては手を付けにくいのではないかと考えた。そこで、XML・XSLT を使用するために必要な最低限の知識を抜き出したオンライン教材を作成することにした。

また、問題を最初から全て自分で解答するのではなく、出題形式を選択形式や穴埋め形式にすることによって簡略化し、ユーザーが手軽に学習できるものにする。選択形式と穴埋め形式の問題を解いていくことで簡単に Web 上で XML と XSLT 学ぶことが出来るため、初心者にとっ

て取り掛かりやすく且つ理解しやすいものになるのではないだろうか。そして、このオンライン教材を作成することで XML と XSLT を学んだことの無い人々が新たに興味を持ち、理解を深めていくきっかけになって欲しいと考えた。

(2) SGML・HTML との相違点と比較

従来の代表的なマークアップ言語である SGML(Standard Generalized Markup Language) や HTML(HyperText Markup Language) と比較しながら、XML の特性を述べていく⁽¹⁾。まず一つ目の相違点は、XML がメタ記述言語だという点である。HTML がや<table>や<p>などの決められたタグのみ使用可能であるのに対し、XML はあらかじめ決められたタグを持たないため自分でタグを定義する必要がある。そのため、与えられたタグに制約を受けることなくユーザーが自由にタグを定義できる。

独自のタグを定義出来る利点は、データ内容を明確に出来ることである。例えば、HTML では<p>頼野温美</p>とする所を XML では<名前>頼野温美</名前>とすることで、タグで囲まれたデータが名前を表していることが分る。<p>は段落を表すタグであり、データそのものを指すタグではない。XML では、データが何を意味するものであるかをタグで明示出来るので、データの構造と内容がどのようなものであるか大変分りやすくなる。

次に、XML は従来構造型標準文書の仕様として開発された SGML に準拠することで、新しい言語でありながら非常に安定した仕様を提供している点である。SGML と違い複雑な仕様を簡略化し、URL 記述やリンク等のインターネットへの対応に優れ、Web 上でも活用可能な仕

様となっている。そして三つ目の相違点として挙げられるのは、「One Source、Multi Use」の性質である。XML はデータのみを記述する言語であるため、データ本体の出力情報やレイアウト定義は一切含まれていない。XML では、データ本体としての XML 文書とレイアウト定義としての XSLT(eXtensible Stylesheet Language Transformations) スタイルシートとを明確に分離することで、情報の再利用・転用性を高めている。一個の XML 文書 (One Source) を、HTML や TEXT など他の形式に変換して様々な用途に活用すること (Multi Use) が出来るからである。

(3) XML と XSLT の現状と基礎を学習する利点

XML と XSLT の基礎の学習の必要性を説くために、まずは XML の現状を述べる⁽²⁾。XML は新しい技術ではなく、基本的な技術として定着している。現在、XML は Web 上で XHTML や RSS、データベースなど様々な方面に活用されている。その他、OS やアプリケーションの設定ファイルに利用されたりもしている。データを XML 文書化する際の最大の利点として、XML に対応した様々なツールや、多くのソフトウェアの利用や連携が可能になる点がある。アプリケーションのデータは通常、特定のアプリケーションにデータを読み込ませて加工や処理を行う。しかし、データが XML 文書であれば特定のアプリケーションが無い場合でも、内容や整合性のチェックには XML パーサが利用出来る。また、データ構造を変換したい場合には、XSLT スタイルシートを使用して複数の形式に変換出来る。これら以外にも、データの操作・分析・加工が可能であり、必要な場合には新たな処理を開発することも容易に出来る。

このように XML が多方面で活用されていることが分かるが、肝心の XML の基本を学習するツールの数が少ない。現在出版されている XML に関する書籍は、XML を使用した他の技術の入門者やさらに発展した XML 技術の専門書などが大半を占める。XML の基本に絞った入門書の数も圧倒的に少ないのである。これらの入門書は 2002 年以前に出版されているものが大部分で、近年、新たに出版されるものは少なくなっている。amazon.co.jp(<http://www.amazon.co.jp/>) で調べた結果、XML 入門書として出版されている書籍の具体的な冊数は、2006 年に 1 冊、2005 年に 2 冊、2004 年に 3 冊であるのに対し、2002 年以前には 9 冊である。2003 年と 2007 年に至っては、1 冊も出版されていない。XML はすでに基本的な技術として浸透しているので、基礎を学ぶというよりも応用を学ぶことになってしまっていると考えられる。

応用した技術を使用するには、まず XML の基本知識を身に付ける必要がある。しかし、それを学習する手段がない。そこで、XML と XSLT の基本的な知識と技術に関するユーザー向けのオンライン教材を作成する必要があると考えた。

(4) 類似サイト・オンライン教材の有無と比較考察

XML と XSLT について用語の解説や基本構造の解説、記述方法などを載せている Web サイトは数多くある。しかし、学習範囲が広く、XML と XSLT の最低限の知識を学ぶサイトは無かった。また、どのサイトも説明文が長く、学習サイトというよりも読み物として作成されているように感じる。実際に文書やスタイルシートを打ち込む作業も、例文を打ち込んで表示させるだけで、確認問題や練習問題などが出題されていない。様々なサイトを見たが現時点では解説、入力式の問題、

問題の解答を含んだ類似サイトやオンライン教材は存在していないと思われる。

『XML-伸び縮みマークアップ言語-』(<http://ykr414.com/xml/index.html>)を考察する。このサイトは、XML や XSLT についての説明が細かな項目で区切られており、説明文も難解な文章が少なく初心者でも理解しやすいように作られている。テキストエディタを起動する所から保存方法まで、XML 文書を作成する一連の流れや、エラーメッセージの対処法などを設けていることも初心者に優しい。また、XML 文書を表示させるために必要なアプリケーションであるパーサ (XSXML3) の説明や、ダウンロード先を提示している点は自分にとって盲点であった。XML を学ぶ以前に必要な知識が本題に入る前に学べるので、安心して学習を始められるのではないだろうか。

しかし、解説や例文を含めて、読んで学習するものが多い。XML 文書や XSLT スタイルシートを実践で覚えるというものではなく、知識として覚えていくだけであるように思える。全ての項目について分りやすく丁寧に解説されており、初歩の学習としては適しているが、問題を設けてユーザーに実際に作業して理解させるようなものが必要だと感じた。読み物として学習するのでは、実際に文書やスタイルシートを作成する際に出来ないことが多い。そのため、目で理解するだけでなく、手を動かして理解してもらう要素が必要である。

また、このサイトは学習内容の幅が広い。基本的な知識や機能だけでなく、XSLT による表示制御についてや DOM(Document Object Model)、SGV(Scalable Vector Graphics) など、さらに一步踏み込んだ項目も盛り込まれている。XML 初心者から経験者まで幅広い利用者を想定したサイトになっているように思える。そのため、後半部分は初

心者にとって理解が難しい内容になっている。

上記のサイトを考察した結果、XML と XSLT の初歩の学習として考えると、XML と XSLT の基本的な知識に絞り込んだ内容のものを作成する必要があると感じた。範囲を絞ることで、学ぶべき内容をピンポイントで学習出来るため、学習してきた内容を覚えやすいのではないだろうか。内容や解説などは簡潔にし、必要最低限のことを教えるようにする。そして、実際に教材を作成する際は、項目毎に細かくページを分けてスクロールが伸びないようにしたい。文章が長く、ページも縦に長いとユーザーに飽きを感じさせてしまうことになるからである。ユーザーを飽きさせない教材にするには、読み物にならないように、ユーザー自身が文書やスタイルシートを作成してもらう形にしなければならない。

(5) オンライン教材にする利点

書籍と違いオンライン教材で学習することは、ユーザーと作成者の両者に利点がある。まず、ユーザー側から書籍の問題点とオンライン教材の利点を述べる。まず書籍は学習内容が幅広いため、深く掘り下げて学習するという点では効果的に学習出来るが、最低限の知識を得ることは容易ではない。そのため、初歩の学習に使用する媒体として書籍を利用するには難しいのではないかと考えられる。基本的な最低限の知識を学習するだけならば、オンライン教材として焦点を絞ったものを作成したほうが理解しやすく役に立つ。さらに、書籍の購入が不要になるため、誰でも手軽に教材を入手出来ることも大きな利点になる。

また、参考サイトの参照や学習に必要なツールのダウンロードに関する点や、実際に授業として XML を学習するという点でも利点がある。まず、書籍を見て参考サイトやダウンロード元へ移動するには、記載さ

れているアドレスを直接打ち込まなければならないので、大変手間が掛かる。オンライン化することでクリック一つで簡単に移動出来るようになり、ユーザーに面倒を感じさせないように出来る。これは、自らが XML と XSLT を書籍を通じて学習してきた経験からそう感じた。次に授業で利用する場合の利点であるが、配布物をオンラインで全て配布出来ることが挙げられる。プリントと違いデータとして配布されるので、配布元にアクセスさえすればいつでも配布物を入手出来る。

次に、作成者側からの利点を述べる。オンライン教材は、内容の差し替えや更新などの変更作業を手軽に行うことが出来る。なぜなら、該当データの内容を修正して新たに Web 上にアップすることで、変更作業が全て完了するからである。書籍の場合は細かな変更点や追加項目が出来ても、新たに改訂版として出版したり刷り直したりと大変手間が掛かるため、内容の修正が簡単に行えないと考えられる。オンライン教材は、作成・変更・配布が統一して行えるので、これらの無駄な手間が省ける。

2 教材の作成まで

(1) 作成時に必要とした技術や知識

本教材は HTML、CSS、Javascript を使用して作成した。Javascript は選択問題と穴埋め問題で使用し、それぞれ解答すると問題の解答ページに移動したり、正誤がアラート表示されるようにした。また、XML と XSLT の教材であるので勿論 XML と XSLT の技術や知識も必要であった。

(2) 教材の構想

教材はまず単なる読み物にならないように注意して作成した。そのため解説が長くなる事は避けて出来るだけ簡潔にし、また、ユーザーが実際に XML 文書や XSLT スタイルシートを入力し、自分自身で結果を表示して確認させるようにした。各項目の解説と、例文を記述した後は必ずそれらをユーザーに実際に入力させて、その結果を確認させるようにする。各項目ごとに確認問題を設け、より理解を深めさせたい。

XML と XSLT を学習するには、XML や XSLT がどのようなものであるかを理解していなければならない。しかし、これらを学習したことのないユーザーには全く知識がない。そこで、はじめに XML や XSLT が何であるかを解説する必要がある。XML 文書や XSLT スタイルシートを作成するために必要な知識を抜き出してユーザーに教えられるようにしたい。

そのために、教材には XML 自体の説明、文書の構成や XML 文書に最低限必要な XML 宣言、DTD、XML インスタンス、XSLT 自体の説明、スタイルシートの構成やテンプレートの書き方、HTML や TEXT 形式への変換方法などを盛り込む。必要な知識を学びながら実際に文書やスタイルシートを入力して作業することで、構成や書き方を身に付けてほしい。最終的な目的は、ユーザーが XML や XSLT が何であるか、またどのように構成され、表示されるのかを理解してもらう事である。

(3) 教材の構成

XML と XSLT がどのようなものであるか、どの様に構成されているかを分りやすく伝える事を目標にサイトを作成した。今回作成した Web ページは解説・問題・解答ページを含めて全部で約 60 ページ (図

1) となった。細かく項目を分けて教材を作成したため、このようにページ数が多くなった。しかし、サイトのファイル構成自体はごく単純なものである。index.html をトップページとし、卒論のテーマや学習対象者、簡単な教材の手引きなどを記述して作成した。トップページにこれらを記述する事でどの層のユーザーに向けて教材を作成したのか、またどのように学習を進めていくと良いのかを明確にした。そして style.css で全ての Web ページのデザインを行い、出来るだけシンプルで見やすいものにした。おおまかなサイト構成は下記の通りである。

- ・ XML とは
- ・ XML 文書の構成
- ・ XML 宣言
- ・ DTD
- ・ DTD の構成
- ・ 要素型宣言
- ・ 属性リスト宣言
- ・ 二つの実体宣言
- ・ XML インスタンス
- ・ XSLT とは
- ・ XSLT の構成
- ・ XSLT の構成 2
- ・ HTML 形式に変換
- ・ TEXT に変換

3 教材の作成内容

(1) XML 編

XML の基本的な知識と技術についてまとめた XML 編は、全 9 項目から構成されている。1 項目めの『XML とは』は、まずはじめに XML とは何であるかを学習するために、XML の概要と主な特徴⁽³⁾を記述したページを作成した。また、これらの解説を簡潔にまとめて記述することで、ユーザーに理解しやすい文章にした。

2 項目めの『XML 文書の構成』では、XML 文書を作成するにあたって必要な要素の名前と文書の基本的な構成を教える。必要な要素をリストで表示し、文書の例を<div>で囲った枠内に表示することによってひと目で構成が分るようにした。ページ末に次の項目に進むためのリンクを貼り、ユーザーに次に進むべきページを指示する。

3 項目めの『XML 宣言』では、XML 宣言とは何であるか、またどのような要素で構成されているかについて解説している。XML 宣言は XML 文書にとって必要不可欠なものなので、より詳しく解説するために XML 宣言自体だけでなく中身である三つの宣言も個々のページに分けて記述した。各ページでは、バージョン宣言、エンコーディング宣言、スタンドアロン宣言の書き方について詳しい解説を記述した。

4 項目めの『DTD』では、DTD 自体の解説と DTD を構成する宣言について記述している。DTD に関するページは、DTD・DTD の構成・要素型宣言・属性リスト宣言・二人の実体宣言によって構成されている。まず、はじめに例文とともに DTD とはどのようなものであるかということと、作成方法について解説することで、おおまかな DTD の役割をユーザーに理解してもらう。

5 項目めの『DTD の構成』では、DTD を構成する要素と DTD の

書き方を説明する。DTD の書き方として、内部サブセットと外部サブセットを解説する。例文を表示して二つの違いを明確にし、用途の違いやそれぞれの特徴を示す。確認問題では外部 DTD に使用する XML 宣言を答えてもらうようにし、XML 宣言の学習ページを参照出来るようにリンクを設けた。

6 項目めの『要素型宣言』では、要素型宣言自体の説明と内容モデル・子要素の出現順序・子要素の出現回数・要素と文字データの混合・要素の優先順位といった細かな項目毎にページを作成した。内容モデルでは、DTD を記述する際に必要な宣言の種類とその意味、記述例を紹介している。子要素の出現順序・子要素の出現回数・要素と文字データの混合・要素の優先順位のページ構成も、同様に作成する。確認問題は入力式の問題と選択式の問題を用いた。

7 項目めの『属性リスト宣言』では要素の属性を定義する属性リスト宣言について説明している。はじめに、属性を構成する要素と構文を記述して、ユーザーにこの宣言がどのようなものであるか理解させる。次に、属性の型と性質について細かく説明文を記述する。

8 項目めの『二つの実体宣言』では、パラメータ実体宣言と一般実体宣言を教える。パラメータ実体宣言は、まず宣言自体の説明文を記述し、それから DTD 内の文字列の置き換えと外部 DTD の取り込みについての解説と例文を記述する。一般実体宣言も同様に、宣言自体の説明と XML 文書の文字の置き換えについての書き方と例文を記述する。同ページに並べて記述することによって、ユーザーに両方の違いを比較してもらい、DTD 内で用いる実体がパラメータ実体宣言であり、内部 DTD として XML 文書で用いられる実体が一般実体宣言であることを理解してもらう。

9 項目めの『XML インスタンス』では、XML 文書のメイン部分である XML インスタンスについての知識を教える。まずは XML インスタンスについて文章のみで説明し、どのような役割を持つか理解してもらおう。そして、どの様に構成されているかを要素・属性・名前の付け方・要素の階層構造・空要素・文字の置き換え・CDATA セクション・コメントとさらに 8 つの項目に分けて説明する。そして最後に、この項目で学習した内容を確認するために XML 文書を作成してもらおう問題を設けた。

まず、『要素』の項目では要素とは何か、どのように記述するかを説明する。それから、確認問題を設けて、学習内容が理解出来ているか確認してもらおう。この段階では、まだ文書を作成してもらうまでに至らないので、知識に関する問題を選択形式にした。ラジオボタンを選択し、解答ボタンを押すと解答ページに飛んで正誤が判明するようにした。ここでは解答だけでなく、なぜその解答になるのかという説明文を簡潔に記述している。

次の『属性』の項目でも同様に、はじめに属性の説明文を記述して、それから書き方の例を表示している。属性は一つだけでなく複数書くことが出来るので、属性が一つだけの場合と複数書かれている場合の二通りの書き方を載せた。確認問題では、属性を定義する際に特に注意しなければならないこの点を取り上げて出題した。

『名前の付け方』では、要素や属性を定義する際の注意点とそれらの名前に使用可能な文字と不可能な文字を説明する。まず、誤りの例を表示し、どの部分が誤りであるか明確にするためにその部分を赤字に変更して分かりやすく示した。次に、名前に使用可能な文字列と不可能な文字列をリストを用いて簡潔に記述する。名前の付け方は、説明文を読

むことで理解出来るものなので、比較的簡単な問題を作成した。

『要素の階層構造』では、XML 文書を記述する際に重要である構造について教える。XML では階層構造で表現するので、階層構造とはどのようなものであるか、その書き方と注意点を例を用いて分かりやすく説明する。階層構造が入れ子状態になっているとはどのような状態を示すのかを、文章で解説するだけでなく、誤った例と正しい例を並べて表示してその違いを明確にした。

『空要素』では、空要素の意味と、開始タグ・終了タグを両方書く場合と終了タグを省略する場合の二通りの書き方を教える。この項目は書き方例を見ると理解出来るものなので、簡潔な説明文と書き方例を表示するのみの内容にした。

『文字の置き換え』では、XML 文書中で、直接記述出来ない文字列について教える。まず、直接記述出来ない文字列を別の文字列に置き換えることを説明し、例を用いてそれがどういうことであるかユーザーに実際に文書を作成させて確認させる。その次に、文字の置き換えの必要性を説明し、定義済み実体の書き方を簡潔に記述して、正しい実体参照の書き方を表示した。ユーザーに実際に文書を作成させて、正しく表示されているか確認してもらう。この項目の問題は、入力式の問題として作成する。フォームに解答を入力してもらい、Javascript によって入力された文字が正しいかどうかアラートを表示させるようにした。

『CDATA セクション』では、CDATA セクションがどのような意味を持つかということと書き方を教える。まず、CDATA セクションについての説明を簡潔に記述し、書き方を表示する。それからユーザーに実際に CDATA セクションを用いた文書を作成してもらい、直接入力出来ない文字がそのまま表示されていることを確認させる。

『コメント』では、内容が比較的簡単であるため、コメントとはどのようなものであるか、どのような場合に使用するか、コメントの書き方を簡潔に記述し作成した。

また、XML インスタンスのまとめとして、これまで学習してきた項目の内容を複数盛り込んだ確認問題を設けた。この問題は、表の内容を基に問題の空欄を埋めて XML 文書を完成してもらうものである。空欄をフォームで作成し、解答を入力してボタンを押すとアラートが表示されて正誤が確認出来る。ここでも Javascript を使用して作成する。また、別に解答ページを設けて問題の下にリンクを貼り、完成した XML 文書がどのようなになっているかを確認させる。

(2) XSLT 編

XML 編と同様に、XSLT の基本的な知識と技術についてまとめた XSLT 編は、全 5 項目から構成されている。まず 1 項目目の『XSLT とは』では、XSLT 全体の説明をしていく。はじめに、XSLT とはどのような機能を持ち、何のために使用するかということ記述して XSLT が何であるかをユーザーに理解させる。次に、XML 文書を XSLT スタイルシートを使用して変換するために必要な前準備の説明をする。文書を HTML や TEXT などの各形式に変換するためには、XSLT プロセッサと呼ばれるツール (msxsl) が必要となるため、ダウンロードページへのリンクを設ける。それから、ダウンロードした後に msxsl.exe を保存する場所と、コマンドプロンプトの起動方法、コマンドプロンプトの設定、カレントディレクトリの設定も一緒に記述しておく。

各設定に関する説明の後に、実際に XML 文書を書いて XSLT スタイルシートが反映されているかをユーザーに確認させるために文書

とスタイルシートの例文を用意した。最後に、これから使用していく `gakusei.xml` という XML 文書を作成させる。この文書は別ページに記述し、直接 XML ファイルにリンクを貼って XML 文書がどのようなものであるかを表示させるようにした。

2 項目めの『XSLT の構成』では、XSLT を構成する要素について説明する。構成を説明するにあたって XML 宣言・スタイルシート宣言・出力宣言の三つの項目に分けて作成する。そして、はじめに XSLT スタイルシート全体の書き方を表示しておき、細分化した各項目へのリンクをリストで表示するようにした。XML 宣言はすでに説明済みなので詳しくは記述しないが、XML 宣言の解説ページへのリンクを設けて参照出来るようにした。次のスタイルシート宣言の項目では、この宣言自体の説明と宣言を構成する要素についての説明を記述している。文章自体は出来るだけ簡潔にし、ユーザーに飽きさせないように作成した。最後の出力宣言の項目でも、スタイルシート宣言と同様の構成で作成した。

3 項目めの『XSLT の構成 2』では、XSLT テンプレートについての説明と書き方を、テンプレート定義・テンプレートの適用・テンプレートの構成・ノードの指定・ルートノード・テンプレートの階層構造といった 6 つの項目に分けて作成した。ここは、一つ一つの説明が短いため `<h4>` で区切り、全て同じページに作成した。どの項目もまずは説明文を記述して、それぞれにどのような意味が定義されているかを理解させる。その後、書き方を表示してユーザーに覚えさせる。

4 項目めの『HTML 形式に変換』では、HTML 形式に変換するためのスタイルシートの書き方と、スタイルシートを記述するにあたって必要なコマンドの説明と書き方を教える。各コマンドの説明後、例文と同じスタイルシートをユーザーに作成させる。それから、`msxsl` の書き方

を説明して、実際に変換作業を行ってもらえるようにする。また確認問題として、穴埋め問題を別ページに設ける。XML インスタンスの確認問題と同様に、Javascript を使用して作成する。ここでも、さらに別に解答ページを設けて問題の下にリンクを貼り、完成したスタイルシートがどのようなになっているかを確認させる。最終的には、ユーザーに問題の XSLT スタイルシートを作成させて、HTML 変換を行ってもらう。

5 項目めの『TEXT 形式に変換』の構成は、教材の統一感を出すために上記の構成とほぼ同様に作成した。

4 作成過程での問題点と工夫した点

(1) 問題点

教材を作成する上で一番の難題として挙げたのは、出題形式の決定と作成方法である。はじめ、HTML で入力項目を作成し、PHP で解答の正誤と変換した文書の表示をさせようとした。しかし XML 文書から HTML や TEXT などの形式に変換するためには、msxsl などの XSLT プロセッサを使用しなければならない。この msxsl の機能を、どのように PHP に取り込むかや、PHP と msxsl をどのように組み合わせて作成すれば上手く動作出来るのかが不明であった。

PHP に関する勉強が不十分であったこともあり、比較的簡単に使用できる Javascript に変更した。そして、出題形式も文書を一から入力して一気に表示させる方法から変更して、知識に関する問題を出題する選択問題と、実際に入力作業を行う穴埋め問題に分けて作成することにした。

作成に必要な技術の問題の解決後、問題のどの部分に Javascript を使用するかという新たな問題が出てきた。知識に関する問題は、入力

フォームを作成して文章を入力してもらおうことも考えたが、文章を打ち込む手間が掛かるためユーザーが解答している途中で飽きてしまうように感じた。選択問題にすることで、簡単に今まで学んできたものの理解度を測ることが出来るのではないだろうか。選択方法も、ラジオボタンとチェックボタンの二通りの方法があったが、二択か三択の少ない選択数にするためラジオボタンを使用することに決めた。

XSLT の部分で使用した穴埋め問題はスタイルシートのどの部分を穴埋めにするかが難しかった。今まで学習してきたものを確認するための問題なので、スタイルシートの構成にとって重要な要素を穴埋めにしなければならない。また文書中にフォームを埋め込むので、スタイルシートの構造が崩れてしまわないように気をつける。入れ子構造が崩れてしまうと、変換する際に上手く作動せずエラーが発生する。しかし、これは HTML で調整することで解決出来る。また、入力フォームの幅や上下の空白を調節し、問題の解答後にひと目で完成した XSLT スタイルシートを確認することが出来るように作成する。

穴埋め問題はアラート表示によって正誤の表示が出現するようにしているので、解答する毎に別ページに移動することがない。入力した項目はフォーム内に残ったままになるので、自分が入力したタグを確認しながら問題を解き進めていくことが出来る。結果的に、このような出題形式にすることで、ユーザーにとって意欲的に取り組める問題が作成できたのではないだろうか。

(2) 工夫した点

(i) サイトデザイン

まずサイトデザインを出来るだけシンプルにして、何がどこに書かれているか分かりやすいように作成した。タイトルや項目は、<h2>と<h4>を使用して項目分けを明確にした。各項目の例文や問題には<div>を使用して、他の文章と線引きした。類似サイトでは、様々な色を使用しており例文と解説の区切りが分からなくなっているものが多い。そのため、教材の色味も統一感を出すために二色に抑えた。重要な点や注意点の文字色を変えることで、より解説を読みやすくした。

最初、全ページにメニューを表示させ、細分化した項目はメニューに加え、主に項目の中に格納していた。しかし、ユーザー自身が学習している項目がどこまで進んでいるのかが分かりにくくなっているように感じた。そのため、全ての項目を一覧表示にして表示している項目の色が変わるようにした。そうすることで、ひと目でどの段階まで学習が進んだかが分かる仕組みに修正した。

(ii) 教材の内容

確認問題や、用語の解説の際に参考ページへリンクで移動出来るようにした。これはあらかじめ、ユーザーが疑問を感じたり、参考になるページが必要であると考えられる箇所に表示した。この様に参考ページへのリンクを設けることで、ユーザーが不明な点をその都度参考ページを参照することが出来るようになり、スムーズに学習を進められると考えたからである。

出題形式も、従来の学習サイトには無い要素を取り入れるために、ラジオボタンを使用した選択形式の問題とフォームを使用した穴埋め形式の問題を設置した。選択問題は Javascript で別の解答ページに移動出

来るようにした。こちらは選択問題と違ってアラート表示にしなかったのは、正解が不正解が表示させるだけでなくその解答に関する解説を載せたいと考えたからである。こうすることで、解答がなぜそのようになるのかを分かりやすくした。

対して選択問題は、アラート表示で正誤を表示させるようにした。これは、空欄に入力した解答の正誤だけが分かれば良いのでこのようにした。そして全体的な解答は別ページに設けて、全体的な XSLT スタイルシートの構成を明確にするようにして解説も加えた。また、XSLT を作成して各形式に変換する一連の流れを記述した。出題ページにこれらを載せることも考えたが、解答を見てユーザー自身に一からスタイルシートを作成してもらうように促しているので、こちらに載せたほうが学習の効率が上がると考えた。

5 おわりに

(1) アンケートによる他者評価

オンライン教材を作成した後、XML 未経験者である人文情報学科の四回生に教材を使用してもらいその評価をアンケートに記述してもらった。アンケートに回答してもらった学生全員が XML と XSLT を知っているという回答した。また、これらを使用したことがあるかという問いに対しては、回答者全員が実際に使用したことはないという回答した。実際に教材を使用してもらった結果は、サイトデザインも見やすく、解説も簡潔で分かりやすいとのことだった。出題形式も、選択問題や穴埋め問題などの形式にすることで初心者にとって取り掛かりやすい教材になっているという意見をもらった。

問題点としては、トップページからこういった順番で学習すべきか

が分からない、例文を実際に動かした結果の画像があるといい、トップページへ戻るリンクがあると助かる、などの意見があった。まず、トップページの各項目へのリンクは、を使用してリストに番号を振ることで解決した。確かに学習する順序が明確に記されていないと、ユーザーが混乱してしまう。次の結果画像の表示については、今後の改善点として最も重要なことであると認識した。この結果画像の有無によって学習の理解度は大幅に変わってくる。そして、各ページからのトップページへのリンクは上部に配置されている XML、XSLT の文字から戻れるように修正した。最終的に、この教材を使用しての XML と XSLT の理解度については、回答者全員から分かりやすいと評価をもらった。

(2) 今後の改善点

今回は初心者を対象としたオンライン教材として、分りやすく且つ見やすいものを作成するという目的の達成はおおむね出来た。しかしより初心者優しい内容にするために、今後の改善点として XML 文書の表示結果の掲載、各項目の例文を画像に変更する、メニューバーのコンパクト化、問題数を増やす、例文や問題の解答や教示結果の掲載、全ブラウザへの対応などが挙げられる。

(i) XML 文書の表示結果の掲載

まず、XML 文書の表示結果を一つしか掲載する事が出来なかった。これは XML 文書がどのようなものであるかを確認するために大変重要なものである。XML 文書を打ち込んでも、それがどの様に表示されるか分らなければ意味が無い。ましてや学習対象者が XML について知識の無いユーザーであるため、XML の構造を学ぶ上で XML 文書の書き方とその文書の表示結果をセットにして教材に載せる事でより優しい内

容になると考える。

掲載した XML ファイルは、ファイルにリンクを貼ることで直接 XML ファイルに移動出来るようにした。画像として教材に載せるよりも、直接 XML 文書の構造に触れることが出来るため、入れ子構造をユーザー自身の目で確認できる。この方法で、他の XML 文書全てを教材に載せたい。

(ii) 各項目の例文を画像に変更する

今回作成した教材は、画像を一切使用せずテキストのみで作成した。そのため、XML 文書や XSLT スタイルシートの書き方例や、問題の解答など<div>を使用して囲い表示している部分を、全て画像で表示させたい。各項目の例文を画像化する理由は、コピー・ペーストを防ぐためである。テキストと違い画像にすると、ユーザー自身が全ての文字を入力しなければならない。目だけでなく、手で覚えることが出来るので、より効果的に学習出来ると考えられる。初心者が XML と XSLT を学習する時に重要な事は、実際にテキストエディタに全ての文書を打ち込み、それぞれの構造やコマンドの意味を覚える事だ。

(iii) メニューのコンパクト化

メニューの長さにも問題がある。全てのページをメニューに一覧として表示させようとする、縦に長くなってしまい、ディスプレイによっては一度に全てを表示することが出来ない。そのためどんなサイズのディスプレイで表示しても、画面内に同様に表示されるように変更したい。しかし、見やすさの点から細分化された項目をメニューから削除することは出来ない。そこで、メニューにカーソルを合わせたときに、細分化された項目を表示するように作成すると良いのではないだろうか。その場合、Javascript や Flash などを使用することでこれらを実現出来

ると考える。

(iv) 問題数を増やす

各項目の問題の数を増やし、それらの解答を全て表示させたい。知識に関する選択問題や、実際に文書やスタイルシートを打ち込んでいく穴埋め問題の確認問題を全てではないが各項目毎に設けた。しかし教材としてはまだまだ数が足りないので、確認問題から発展した練習問題や、学習してきた内容を総括した最終問題を作成しなければならない。ユーザーにとって学習する上で重要なことは、繰り返し学習することである。

(v) 例文や問題解答の表示結果の掲載

手順や解説を文章で記述しているのみで、全ての例文を実際に動かした結果を表示していなかった。同様に、XSLT を使用した穴埋め問題の方で、問題となっている XSLT スタイルシートの解答は載せているがそのスタイルシートを使用して変換した HTML 文書や TEXT 文書の表示結果などを載せていなかった。ただ解答を載せるだけでなく、XML 文書や XSLT スタイルシートの作成後の一連の作業と最終的な結果の表示を加えることが重要であると考え。そうすることで、より理解の深まる教材が出来るのではないだろうか。

(vi) 全ブラウザ対応にする

そして、IE 以外のブラウザでの表示であるが、FireFox 上で教材を使用した。サイトデザインは IE と同様に表示されるが、Javascript を使用した穴埋め問題にエラーが出た。穴埋め問題はアラート表示によって正誤が分るもので、FireFox で見てみるとこのアラート表示にエラーが出ている。Javascript に関しては勉強不足のため改善方法が今の所分らないが、今後全てのブラウザで同様に機能するように改善する。

(3) 自己評価

今回オンライン教材の作成を終えて、実現できなかった箇所が多々あったが、当初の目的であったユーザーにとって分りやすく且つ見やすいサイトの作成はおおむね達成できたものとする。しかしオンライン教材という性質上、どのブラウザでも同様に表示し、機能するものを作成しなければならない。そうでなければオンライン教材にする意味がない。このような問題が起きたのは、Javascript の勉強不足が原因である。教材を作成する以上、必要となる技術を最低限学習する必要があると感じた。

また教材としての面では、基礎的な知識や例文は盛り込むことが出来たが、出題数の少なさと内容に問題があるとする。ユーザーの学習方法としては、繰り返しの学習が効果的であるといえる。そのため、確認問題だけでなく、確認問題から発展した練習問題、最終的な理解度を問う最終問題の作成など数と種類を増やすことが重要である。実際に多くの人の学習に役立ててもらえる教材にするには、まだまだ改善が必要である。最終的に、授業や長期休暇中の課題として活用出来るレベルのものを作成したいとする。

XML と XSLT の初歩的な学習は、最低限必要な知識や技術を学ぶことが重要である。簡潔な説明による各項目の解説を読み、問題を解くことで知識や技術が身に付いていくのではないだろうか。書籍と違い、オンライン教材は学習範囲を絞り込んで作成することが出来る。そのため XML 自体の入門書の数が少ない現在において、初心者向けの学習ツールとして活用していくべきだと考える。

注

- (1) 山田祥寛 『XML 辞典』 2006、 p.19 より一部引用
- (2) 『@ IT-アットマーク・アイティ-』(<http://www.atmarkit.co.jp/>)
から一部引用
- (3) 『XML-伸び縮みマークアップ言語-』(<http://ykr414.com/xml/index.html>)
から一部引用

文献表

松尾泰

2006 『改訂 XML 入門』 株式会社 SCC

山田祥寛

2004 『XML 辞典』 翔泳社