

Excel, Word 自動採点システム HITs の構築と運用 Construction and Operations of an Automated Marking System for Excel and Word files

岩田 員典（愛知大学経営学部）

功刀 由紀子（愛知大学経営学部）

齋藤 毅（愛知大学経営学部）

谷口 正明（名城大学総合数理教育センター）

長谷部 勝也（愛知大学名誉教授）

松井 吉光（愛知大学法学部）

古川 邦之（愛知大学経営学部）

要旨

本学名古屋校舎では2006年度の秋学期から Microsoft Office を利用して学生が作成した課題を自動採点し、その結果を保存する統合システムとして HITs の運用を進めてきた。HITs は Web サーバ上で動作する自動採点システムの総称であり、インターフェースやデータベースの処理には PHP を主に使用して記述している。課題の採点には、Python, Perl を用いている。本論文では HITs の概要について述べる。さらに、Excel, Word の採点の仕組み・機能を説明し、本システムで使用しているコンテンツについても説明する。

キーワード：e-Learning, 自動採点, 情報リテラシー教育, 情報教育

1. はじめに

本学名古屋校舎では2006年度から1年生の情報リテラシーレベルを向上させることを目的として情報リテラシー科目を設定している。2006年度の春学期においては、本科目は完全な自学自習形態とし、市販教材を用いた課題提出による単位取得という方式を採用した。しかしながら、従来の e-Learning に関する市販

教材は、画面の内容を読解する講義型であり、課題試験も知識を択一形式で出題するのみである。そのため、受講者としては興味が湧かず、その場で正答すればよいという一過性の学習になりがちであった。そこで、市販教材では情報リテラシーレベルの向上は困難であると判断し、対面型演習形式で自学自習のできる e-Learning 教材の独自開発に取り組んだ。その結果として、Microsoft Office

を利用して学生が作成した課題を自動採点し、その結果を保存する統合システムとしてHITs (Highly Interactive Training system)を開発・運用してきた。本稿では統合システムHITsの概要について述べ、Excel, Wordの採点の仕組み・機能を説明する。

本論文は次のように構成される。次章では関連システムについて述べる。次に、3章でHITsの概要について説明する。そして、4章でExcelの採点エンジン・コンテンツ、5章でWordの採点エンジン・コンテンツについて説明する。最後に6章でまとめと今後の課題について述べる。

2. 関連システム

HITsについて述べる前に関連が深いと考えられるMicrosoft Officeのファイル自動採点ソフトについて述べる。このようなソフトは以下の2種類に分類できる。

- Flashなどを用いて擬似的なOffice環境を作り出し、定められた操作を守っているかどうかを採点するもの。
- Microsoft社とライセンス契約を行い、与えられた仕様を基にしてファイル内部の解析を行うもの。

前者は、擬似的なOffice環境ということもあり、操作方法などを逐次的に採点しているに過ぎない。また、その環境で定

められた手順に従って操作しなければ、正解とならないことが多い。例えば、マウスを使った操作は正解になるけれども、ショートカットキーを用いた操作は不正解となることがある。一方、後者は、MicrosoftあるいはMicrosoftと有料でライセンス契約を行った者が、Officeファイルの中身を解析・採点できる自動採点エンジンにより採点を行うものである。これらは商用のソフトとして販売されており、採点エンジンを編集したりそれに機能を追加したりすることは禁じられている。また、これらのソフトでは教員が新規の問題を作成し、それを利用することはできないなど拡張性に乏しい場合が多い。そして、問題の数が少ないことに関連して、これまでの自動採点ソフトは、受講者が学習を終えた際の実力テストや検定テストとして用いられていた。

3. 自動採点システム HITs の概要

HITsはExcel, Wordを利用して学生が作成した課題を自動採点し、その結果を保存する統合システムである。春学期にHITsを利用して情報リテラシー入門を受講した学生数は900名弱であり、同時に授業で利用している最大人数は110名である。また、秋学期の情報リテラシー応用では約480名の学生が利用している。本章ではHITsの概要について述べる。

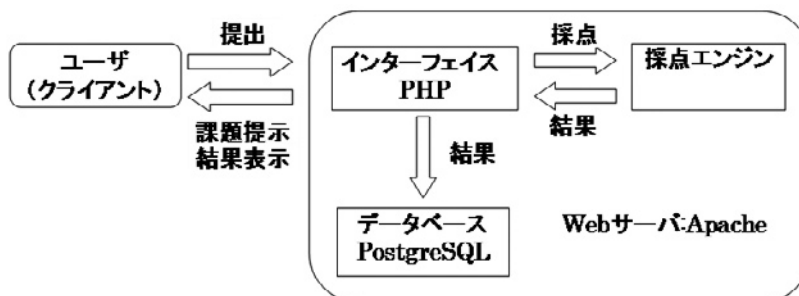


図1：HITsの概要

3.1 課題提示と採点の概説

HITsの課題提示と採点の仕組みを以下に示す。

1. ユーザは HITs に Web ブラウザ 経由でアクセスする（図2）。
2. Web ブラウザ に表示された課題を適宜ダウンロードする（図3）。
3. ユーザは問題を PC で解答を作成する。
4. 解答をアップロードする（図4）。
5. HITs はアップロードされたファイルを自動採点エンジンによって採点し、その結果をデータベースに保存する。
6. また、採点結果をインターフェースを通じてブラウザに表示する（図5）。

練習問題 21

エクセルを使い、以下の表の空欄を四則や計算式を使って計算しなさい。
ただし、以下のファイルをダウンロードして計算を行ない、xlsx ファイルとして保存した後、提出しなさい。
注意：行や列を挿入して表の体裁を変えると正常に採点されません。

練習問題21(ダウンロード)

	A	B	C	D	E	F
1	練習問題 21					
2	あるレンタルビデオ屋では、一枚のDVDを280円で貸してくれる。					
3	しかし、期日までにDVDを返さないとい日当たり280円の遅延金を取る。					
4	さて、以下の表で遅延金はいくらになるか、計算しなさい。					
5	(ヒント: まず、B9に「F=B9+280」と入れる。後は、下にオートフィルすればよい。)					
6						
7	遅延日数(日)	遅延金(円)				
8	1	280				

図3：課題のダウンロード

問題の提出

提出するファイルを選んで、提出ボタンを押してください。

提出するファイル:

図4：課題の提出

HITs -- Dynamic e-Learning ログイン

現在の時刻: 2009年9月16日(水)12時05分20秒

学籍番号(I):

パスワード(P):

図2：ログイン画面

解答欄

	A	B	C	D	E	F
9	種類	単価	個数	金額	割合1	割合2
10	うめ	90	95	8550	11.4%	8.8%
11	おかめ	110	128	14080	15.4%	14.4%
12	ツナ	120	86	10320	10.3%	10.6%
13	こんぶ	100	134	13400	16.1%	13.7%
14	五目	150	72	10800	8.6%	11.1%
15	明太子	130	112	14560	13.4%	14.9%
16	鮭	120	57	6840	6.8%	7.0%
17	たらこ	110	63	6930	7.6%	7.1%
18	山ごぼう	150	41	6150	4.9%	6.3%
19	野沢菜	130	45	5850	5.4%	6.0%
20		合計	833	97480	100.0%	100.0%

おめでとう、全問正解です!Excelの練習問題49はこれで合格です。

図5：採点例

自動採点エンジンの詳細については次章以降で説明する。

3.2 HITsと関連システムの相違

HITsと前章で説明した関連システムとの違いは次のようにまとめられる。

- 定められた手順ではなく完成した解答を採点。
- 管理部分と自動採点エンジンは独立なため、変更や機能の追加が容易。
- 教員による問題追加・編集が可能。
- HITsに多くの問題を追加することにより、HITsは実習内容に対応した言わばドリル的な使い方が可能である。

4. Excelの採点エンジン

最初のExcelの採点エンジンはHTML形式で保存されたExcelファイルを採点していた。その後、Excel 2003のバイナリ形式を採点できるように変更した。そこで、本章ではまずHTMLでの採点方式について説明し、続いてバイナリ形式の採点方式について説明する。

4.1 HTML形式の採点

Excel 2003をHTML形式で保存して採点する際にはPerl 5.8.6を使用した。採点の基本方針としては関数（もしくは計算式）を使用しているセルについて採点をおこなう。

4.1.1 セルが持つ情報

HTMLとして保存されたセルは次のような情報を持つ。

- 数字だけのセル

```
<td align=right x:num>52</td>
```

- 関数を使用しているセル（図6）

このように「x:fmla」属性を持つセルの情報に着目することで、採点が可能になる。

4.1.2 採点方法

4.1.1の情報を利用し以下の手順で採点をする。

1. 属性「x:fmla」が存在するか否かにより、計算式を使用しているかどうか判断する。
2. 属性「x:num」が示す値を調べることで、計算後の値が合っているかどうか判断する。
3. <td></td>タグで挟まれた値を調べることで、書式があっているかどうか判断する。尚、書式とは以下のような条件を示す。
 - （ア）表示桁数
 - （イ）パーセント表示
 - （ウ）金額表示

これらの条件に従い、解答例から採点基準ファイルを作成する。採点基準ファイルは図7のような情報を持つ。このファイルにおいて「2:5」は、どのセルの情報であるかを示し、その次の行が解答例となる。尚、「---」はセル情報の区切りを示す記号である。

```
<td align=right x:num x:fmla="=SUM(C3:E3)">125</td>
<td align=right x:num="41.666666666666664" x:fmla="=AVERAGE(C3:E3)">41.667</td>
```

図6：関数を使用しているセルの例

	セルの番号
2:5 ←	2→行番号
	5→列番号 (F 列)

```
<td align=right x:num x:fmla="=SUM(C3:E3)">125</td>
```

2:6

```
<td align=right x:num="41.666666666666664" x:fmla="=AVERAGE(C3:E3)">41.667</td>
```

図7：採点基準ファイルの例 (HTML)

図7の情報に従うと、「F2」(2:5で指定)のセルは計算式を使用し、計算後の値が「125」であれば正答とする。この際に「x:num」属性は値を持たないため、調べる必要がない。一方、「G2」(2:6で指定)のセルを採点する場合は、計算式を使用しているかを調べた後、「x:num」で指定されている値「41.666666666666664」と解答が等しいか調べ、さらに書式適応後の値「41.667」と解答が等しいか調べる。

前述の例で示したとおり、採点においては計算式を使っておればよく、どのような計算を行っているかは問わない。これは同値となる計算が複数存在することから、解答者の選択肢を狭めないためである。例えば「G2」(2:6で指定)のセル

の場合、次のような計算が考えられる。

- =AVERAGE(C3:E3) (解答例)
- =SUM(C3:E3)/COUNT(C3:E3)
- =SUM(C3:E3)/3
- =(C3+D3+E3)/3

ただし、この採点方式の場合

「=41.666666666666664」と入力されても正答となるが、このような解答は手間がかかるだけなので解答者は選択することはないと推測される。

4.2 Excel 2003バイナリ形式の採点

Excel 2003をHTML形式で保存して採点する方式は、解答者が誤ってバイナリ形式(拡張子.xls)で保存してしまう

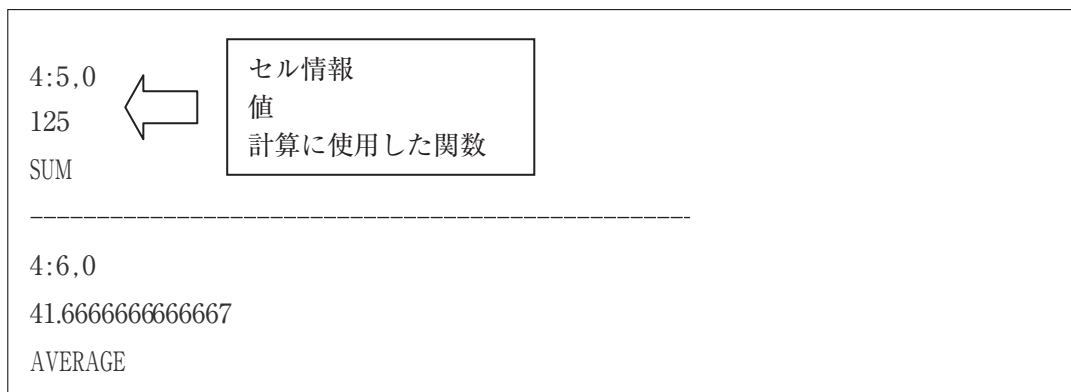


図8：採点基準ファイルの例（バイナリ形式）

と採点できないという問題があった。そこで、Spreadsheet-ParseExcel¹⁾を利用することにより Excel 2003 のバイナリ形式で採点出来るようにした。採点の基本方針は HTML 形式と同じく関数（もしくは計算式）を使用しているセルについて採点をおこなう。また、マクロを使うことでグラフの採点にも対応した。

4.2.1 取り出せる情報

Spreadsheet-ParseExcel とマクロを利用することで Excel 2003 バイナリ形式のファイルから取り出せる情報は以下の通りである。

- セルの値
- セルに入っている値の型
 - Text
 - Numeric
 - Date
- 値の型が Numeric の場合は、どのような式を使っているか
- グラフの種類（マクロによる）

- グラフの項目軸ラベル（マクロによる）
- グラフに表示されている値（マクロによる）

4.2.2 セルの採点方法

4.2.1 で示した情報を利用し、解答例から採点基準ファイルを作成する。採点基準ファイルは図8のような情報を持つ。4.1.2 の例と異なるのは、セルの情報の後に「採点モード」に関する情報があることである。また、セル情報の次の行はそのセルが持つ値であり、さらに次の行は計算に使用された関数を示している。採点モードは以下の4種類ある。

- 採点しない
 - 式を使っている値が正しいか
 - 指定された関数を使い値が正しいか
 - 値が正しいか（式は使わなくてもよい）
- このように HTML のときと大きく異なるのは、指定された関数を使っているかどうかとも採点出来ることである。これにより、例えば HTML を使用した方式では

平均値を求めるのにどのような式を使ってもよかったが、本方式ではAVERAGE関数を使うように指示することが可能となった。

4.2.3 グラフの採点方法

グラフを採点するために、グラフ情報を読み取りそれをセルにデータとして保存するマクロを作成した。このマクロはファイル保存時に実行され、4.2.1で示し

たグラフに関する情報が記録される。この情報は200行目以降に背景色と同じ色で記録する。そのため、解答者はこの情報の存在に気づきにくいといえる。

グラフの例と保存された情報の例をそれぞれに図9、図10に示す。尚、図10の情報は本来は背景色と同じ色（白色）のため見られないが、本稿では黒色にして見られるようにしている。

このようにマクロを利用することでグ

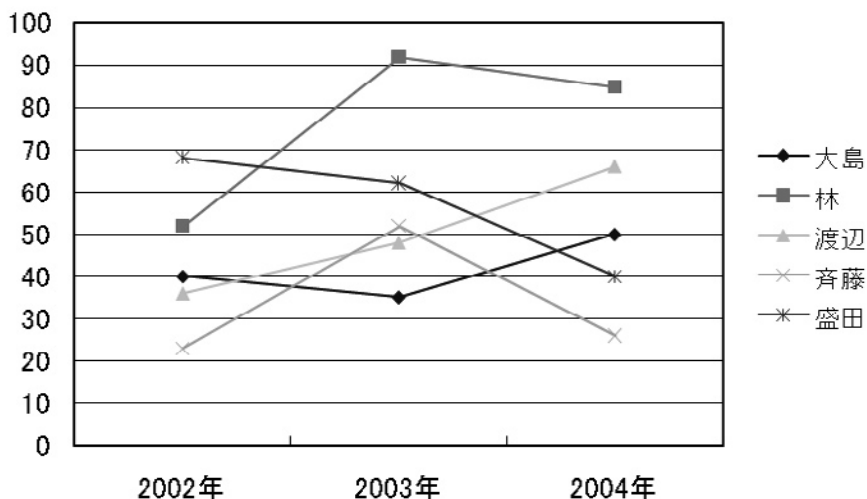


図9：グラフの例

200	Chart Type	4		
201	Series	5		
202	GapWidth	100		
203	X Values	2002年	2003年	2004年
204	Series1	40	35	50
205	Series2	52	92	85
206	Series3	36	48	66
207	Series4	23	52	26
208	Series5	68	62	40

図10：グラフ情報の例

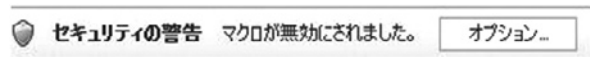


図 11：マクロによるセキュリティの警告



図 12：警告の解除

ラフの採点が可能になったが、セキュリティ面から警告が表示され(図 11)、それを解除する(図 12)ことによって採点ができるようになるという問題点は残る。

5. Word の採点エンジン

2006 年夏ごろの HITs の開発初期には、Word ファイルの採点は、Excel と同様に Word が出力する HTML を採点していた。Word の出力する HTML は、標準的な HTML であるため採点は容易であったが、文字列の装飾や段落の体裁を表すタグが各所に入っているため可読性

は高くなかった。また、単なる HTML が Word の多様な文章表現能力を再現するには役不足である面は否めなかった。

この当時、Word のバージョンは 2003 であった。Word 2003 からは、Word で作られたファイルを XML 機能のフルサポートが行われており、また、その仕様も公開されていた³⁾ ため、私達は将来性を考え、直ちに XML ファイル(拡張子が .xml のファイル)を採点するプログラムを開発することに着手した。

その後、Office のバージョンは 2007 に上がり、その他の Office 系ソフトの XML 化も相まって、Office Open XML

File FormatsとしてECMAから公開されるに至った。Wordに関しては、一部の機能が追加されたものの、ほぼWord 2003のXML文法を踏襲していたため、初期に作ったプログラムを基本的にはそのまま使い続けている。

この章では、Word XMLの構造の簡単な例を説明し、次に採点の方法についてまとめる。

5.1 Word XML形式の概要

Word XMLの構造はHTMLの構造によく似ている。これを簡略化したものを図13に示す。w:bodyは本文を表す。w:pは段落を表す。通常、w:pは一つのWord文章に段落の数だけ表れる。(なお、Wordでは、改行から次の改行までの間が一つの段落である。)段落の体裁(中央揃えなど)はw:pPr(paragraph property 要素)の属性値として指定される。その有効範囲は、その段落内であ

る。また、装飾付き文字列を表すのはw:r(run 要素)である。その中にはw:tという要素があり、この中に文字列が格納される。文字列の装飾は、w:rPrという要素の属性値で指定される。当然ながら、w:rは一つの段落内に複数存在する。また、一つのファイルには一つ以上の節があり、例えばページ設定などの節の設定に関する情報はw:sectPrに格納される。これらの要素には開始タグと終了タグがあり、例えば、<w:body>と</w:body>の間に<w:p>～</w:p>などの子要素が記述される。

例えば、「新工場開設のお知らせ」と書き、フォントサイズを12ptにして下線を引き中央揃えをしたとしよう(図14)。

このファイルをXML形式で保存して、段落要素を抜粋したものが図15である。<w:jc w:val="center"/>は中央揃えであり、<w:sz w:val="24"/>はフォント12pt、<w:u w:val="single"/>は一重下線を表している。

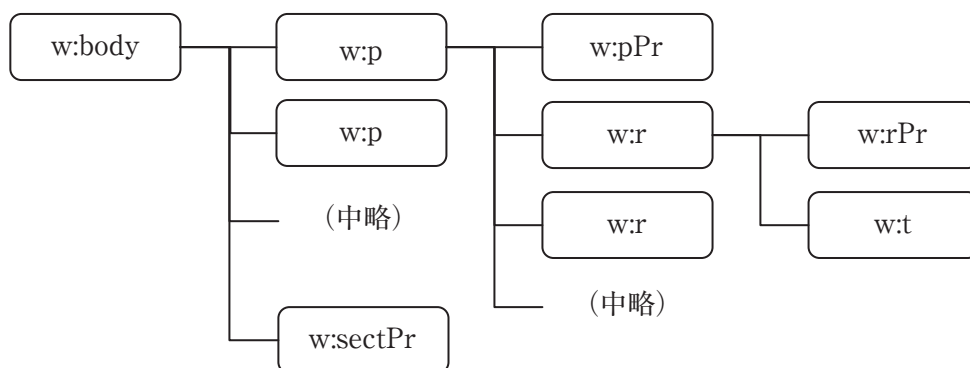


図13：Word XMLの構造



図 14：Word の問題の一例

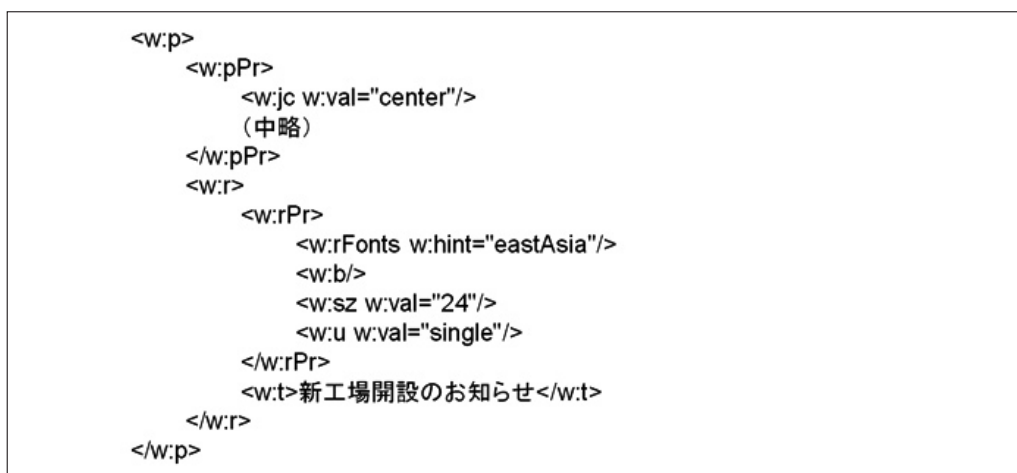


図 15：Word XML の一例

5.2 Word XML 形式の概要

ユーザが指定された操作を行っているかどうかは、ユーザが提出した XML ファイルの文法解析を行えばよい。Word XML の採点には Python 2.4.4 とこれに付属する XML parser (解析器) である Expat モジュール⁴⁾を使用した。

私達は、採点プログラムで次のような手順で採点を行った。

1. 採点対象となっている文字列を探す。
2. その文字列に適応されている XML 要素や属性を探す。
3. これらが採点基準ファイルと一致す

るかどうかを見る。

図 14 図 15 の問題ではフォントサイズが 12ptであることを次のようにチェックしている。

1. 最初に本文中にある w:t 要素の中から「新工場開設のお知らせ」という文字列を探す。
2. そこを起点として XML tree を 1 段上がり、w:r 要素以下に w:sz 要素が存在するかどうかを探す。また、その属性値が、w:val="24" (12pt を意味する) かどうかをチェックする。
3. これらが満たされている場合正解とする。満たされていない場合は不正

```

text 新工場開設のお知らせ
ife
    print 正しい文字列が見つかりませんでした
else
    upwd 1
    elem w:rPr
    elem w:sz
    ife
        print フォントの設定をしていません
    else
        attr w:val = 24
        ife
            print フォントサイズを間違えています
        else
            (中略)
        endif
    endif
endif
endif

```

図 16：採点基準ファイルの一例

解とする。正解あるいは不正解の時に返すメッセージは採点基準ファイル（CTL ファイル）に記しておく。ここで図 16 に上記の採点基準ファイルを示しておく。この中で用いられている各コマンドは採点プログラムで定義しており、それぞれの意味は以下のとおりである。

text: XML ファイルの中から指定された文字列を探す。

ife: if error の意味で、上記の検索に失敗したときの処理を記述する。

else: 上記の検索に成功したときの処理。

print: 画面にメッセージを出力する。

upwd: XML tree を一段上がる。

elem: 指定された要素 (element) を探す。

attr: 指定された属性 (attribute) を探す。
 なお、ここでは省略したが、下線と中央揃えの採点に関しても同様のコマンドを用いる。

5.3 Word XML 採点の問題点

上記の Word XML の採点方法はこれまでのところうまく機能しているが、いくつかの問題点にも直面しているため、今後の開発のためにもここにまとめておく。

文字列の分断

デフォルト設定で、Word はスペルチェックやユーザの編集履歴などの情報をファイル中に保存する。例えば、ある

文字列の途中で文字を挿入した後、それを削除した場合には、その箇所にユーザが編集したというマークがつき、その文字列要素(w:t)は分断されてしまう。また、スペルチェックに引っ掛かるような文章を書いた場合に、デフォルト設定では自動スペルチェックが働き、赤または緑色の波線が表示される。Word XMLではこの波線が引かれた部分にスペルチェックに引っ掛かったことを示す要素が挿入されてしまい、同時に文字列要素(w:t)が分断されてしまう。

これらは、これまでの私達の採点手法になじまないものであり、採点が不可能（あるいは不正確）になることを意味する。このため私達は、スペルチェックに関するオプションと編集履歴に関するオプションを情報メディアセンターにOFFに設定してもらって運用を行った。

採点基準ファイルの手動作成

これまでの採点方法では、採点基準ファイル(CTLファイル)を手書きで書かなければならなかった。これは主にWord XMLの構造が複雑であり、かつ、採点にはさまざまな要素や属性が必要だったためである。このため、問題作成者は、Word XMLの文法をある程度知っていなければならなかったし、採点基準ファイルの書き方に習熟しなければならなかった。これは問題作成者にある程度の訓練が必要であるということを意味し

ており、問題数の増加に対する足かせとなっており、同時にHITsの一般への普及の妨げとなっている。

少なくともWebベースで採点基準ファイルを簡単に生成できるような仕組みが必要であり、このような方法が実現できるように努力中である。

旧バージョンへの対応

これまで私達は、WordファイルをXML形式で保存したものを採点してきた。ただし、Word 2003が出力するXMLとWord 2007が出力するXMLは要素名や属性名などがわずかに異なっていた。このため、両者のどちらとも採点できるようにするためには、採点基準ファイルを2種類用意する必要があった。また、Word 2002以前のWordはXML形式でのファイル保存ができなかったため、採点することができなかった。

Word 2007からは、Word文書は標準でdocx形式で保存することとなった。このファイル形式は、Word 2007のXMLファイルを目的別にいくつかのファイルに分けて、それをZIP圧縮しただけのものである。また、マイクロソフトからWord 2000以降のバージョンでdocx形式をサポートする（読み込み、書き込み、保存ができる）パッチが配布されている。

したがって、docxファイルの採点を行うことは、これまでの方法の自然な延長

で行うことができ、かつ、Word 2000以降のファイルも docx 形式で保存させれば、一つの採点基準ファイルで採点できることを意味している。このように大きな長所があるため、現在これに対応できるように鋭意努力中である。

6. まとめと今後の課題

本稿では統合システム HITs の概要について述べ、その関連システムとの違いを説明した。そして、Excel、Word の採点の仕組み・機能をそれぞれ説明した。Excel の採点では最初は HTML 形式に対応させていたが、それを Excel 2003 のバイナリ形式対応に更新した。この更新に伴い、採点時に計算に使用する関数を指示できるようにするとともに、マクロを利用したグラフの採点も可能になった。

今後の課題は以下の通りである。

● Microsoft Office 2007 への対応

Office 2007 では Excel、Word ファイルは OpenXML 形式²⁾で保存されているため、この形式に対応することでより柔軟な採点方式が可能になるはずである。

● コンテンツの充実

システムの運用開始時から比較すると、かなりのコンテンツが用意できたが、さらなる充実が必要である。

● 配布の仕組みの確立

本学だけでの利用から学外機関にも

利用してもらうために、システム全体をインストールした CD (DVD) ブート Linux などの準備が必要である。

また、採点エンジンを SCORM 対応にすることで、各種 LMS に対応させていくことも検討したい。

謝辞

本稿の一部は私立大学等経常費補助金特別補助(教育学術データベース)によった。

注・文献

- 1) Spreadsheet-ParseExcel, <http://search.cpan.org/~jmcnamara/Spreadsheet-ParseExcel-0.54/lib/Spreadsheet/ParseExcel.pm>
- 2) Standard ECMA-376 Office Open XML File Formats, <http://www.ecma-international.org/publications/standards/Ecma-376.htm>
- 3) Office 2003 XML リファレンススキーマ, <http://www.microsoft.com/japan/office/previous/2003/xml/default.mspx>
- 4) <http://www.python.jp/doc/release/lib/module-xml.parsers.expat.html>